3D INTERFACES FOR SPATIAL CONSTRUCTION

Thesis by
Steven Schkolne

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California
2004
(Defended October 2003)

abstract

It is becoming increasingly easy to bring the body directly to digital form via stereoscopic immmersive displays and tracked input devices. Is this space a viable one in which to construct 3d objects? Interfaces built upon two-dimensional displays and 2d input devices are the current standard for spatial construction, yet 3d interfaces, where the dimensionality of the interactive space matches that of the design space, have something unique to offer.

This work increases the richness of 3d interfaces by bringing several new tools into the picture: the hand is used directly to trace surfaces; tangible tongs grab, stretch, and rotate shapes; a handle becomes a lightsaber and a tool for dropping simple objects; and a raygun, analagous to the mouse, is used to select distant things. With these tools, a richer 3d interface is constructed in which a variety of objects are created by novice users with relative ease. What we see is a space, not exactly like the traditional 2d computer, but rather one in which a distinct and different set of operations is easy and natural.

Design studies, complemented by user studies, explore the larger space of three-dimensional input possibilities. The target applications are spatial arrangment, freeform shape construction, and molecular design. New possibilities for spatial construction develop alongside particular nuances of input devices and the interactions they support. Task-specific tangible controllers provide a cultural affordance which links input devices to deep histories of tool use, enhancing intuition and affective connection within an interface. On a more practical, but still emotional level, these input devices frame kinesthetic space, resulting in high-bandwidth interactions where large amounts of data can be comfortably and quickly communicated.

A crucial issue with this interface approach is the tension between specific and generic input devices. Generic devices are the tradition in computing — versatile, remappable, frequently bereft of culture or relevance to the task at hand. Specific interfaces are an emerging trend — customized, culturally rich, to date these systems have been tightly linked to a single application, limiting their widespread use. The theoretical heart of this thesis, and its chief contribution to interface research at large is an approach to customization. Instead of matching an application domain's data, each new input device supports a functional class. The spatial construction task is split into four types of manipulation: grabbing, pointing, holding, and rubbing. Each of these action classes spans the space of spatial construction, allowing a single tool to be used in many settings without losing the unique strengths of its specific form. Outside of 3d interface, outside of spatial construction, this approach strikes a balance between generic and specific suitable for many interface scenarios.

In practice, these specific function groups are given versatility via a quick remapping technique which allows one physical tool to perform many digital tasks. For example, the handle can be quickly remapped from a lightsaber that cuts shapes to tools that place simple platonic solids, erase portions of objects, and draw double-helices in space.

The contributions of this work lie both in a theoretical model of spatial interaction, and input devices (combined with new interactions) which illustrate the efficacy of this philosophy. This research brings the new results of Tangible User Interface to the field of Virtual Reality. We find a space, in and around the hand, where full-fledged haptics are not necessary for users physically connect with digital form.

dedication

This thesis is dedicated to my grandfather, Edmond Cecil Bloch, who helped me get started with computers when I was a child. An accomplished research scientist who loves to understand, his broad mind has always been an inspiration.

acknowledgments

It is really some small miracle, looking back, that I came to Caltech being who I was and managed to produce this book that lies before you, not to mention the research behind it. This is the result of the consistent support of a whole community of people, but a lot of it — say about half of what it took to get this thesis finished is due to the guidance and support of Peter Schröder. I agreed to be his student because he wore such a nice jacket for interview day, and I guess I had my priorities in order, because he proved to have a love of aesthetics and humanity on top of a stellar history of researching the mathematical foundations of computer graphics. What I could not have predicted was how far he went in supporting me, even when it clearly did nothing to help his tenure case, his funding, or any of the shallow things that build a career in our career-centric universities. I have known many graduate students, and never have I witnessed one with as much freedom as I found at Caltech, and for this Peter, I cannot thank you enough.

I have been fortunate to have had mentors — Al Barr, Hiroshi Ishii, Miltos Manetas, and an interesting figure who goes by davidkremers, all of whom have given to this process, my process, without ever asking for a return favor. When long conversations were necessary, they were all there for me, taking me further than I thought I ever needed to go. My thesis committee — consisting of Al, Peter, Hiroshi and also Erik Winfree and Pietro Perona, showed me a side of knowing which I didn't know was. They cared, for no other reason than a pure intellectual integrity, which I have tried my best to accumulate during our meetings and pour into this document. Whatever clarity you find in this thesis is due to the efforts of these people.

Writing a PhD is not about a lab, it is about the world — history, culture, and space are the true determinants of a thesis' worth. And I must thank my loving friends, too numerous to mention, and my indefatigable family for educating me in the life which any worthwhile research aims to effect. Without you I would have surely crumpled, fallen, and faltered long before the finish line.

Special thanks to Amera Rizk for designing the look of this book.

contents

0 audiences

who should read this book, and what they should expect

SPATIAL CONSTRUCTION INTERFACE DEVELOPERS will see a decomposition of the process of 3d digital creation. I focus on the relationship between physical input and digital action, with a special attention to the relationship between body space and model space. Various examples of these mappings, both successes and failures, provide a deeper understanding of ways to phrase virtual space around users. From this emerges a general framework for effective interaction that allows a wealth of intricate spatial maneuvers with a few tangible input devices. These readers should probably start directly with Part II, and read Part III as their interest warrants.

INTERFACE RESEARCHERS AND DEVELOPERS that do not focus on spatial construction will both learn about bringing Tangible UI to 3d, and moreover see certain general issues that are particularly evident in this domain. Extending Tangible UI to 3d provides a unique opportunity to tie the body directly with digital form. The strength of this mapping is described by the concepts of kinesthetic framing and direct mimicry. The use of continuous inputs, typically thought to be a strong component of 'natural' interaction, was found to be much less successful than direct, discrete, input. The trade-offs between custom and general-purpose interface design are particularly evident in the design paths taken in this text. Despite their promise, immersive systems and virtual reality have traditionally been quite difficult to use, and this text shows that a frame-centric approach to 3d interface is more useful than point manipulation. This group of readers might want to start directly with Part III to get a better understanding of the interface issues at hand. After skimming this section you might see more of the issues involved in the in-depth design seen in Part II.

THE VIRTUAL REALITY COMMUNITY A large amount of research in virtual reality focuses on sensing techniques and user presence in large environments. This text focuses on a small semi-immersive space, all of it within arm's reach, where users can design cars, ballerinas, and double helices, all with a few tools. While this is not Virtual Reality in its purest form, it demonstrates a richness of interaction that can be had without precise tracking. My interfaces are built on shoddy, poorly calibrated trackers. I have grown lazy about calibrating them because the interactions in this book do not require accurate tracking. While the need for better sensors, and the desire to reach the holy grail of pure immersion in a synthetic space are still strong within the VR community, this text shows applications where some real results can be had, results which could have a large impact on design, manufacturing, and the arts. While these results will not show the Virtual Reality community a full reality, hopefully this core of successful interaction can be grown to these ends.

This text asks the Virtual Reality community to look closely at the process of 3d creation. I do not feel that my interfaces are ideal, in all hope there are simpler, more direct, more emotive interfaces waiting right around the corner. The message is not to use my interfaces as much as to concern yourself with the application of spatial construction, for it is this task that appears to be the killer app of Virtual Reality technology. This text demonstrates that these tools rest on basic interaction elements, more sophisticated tools are a matter of recombining these elements, and that an improved interaction for all of spatial design is within reach. From this basis, a fuller, richer Virtual Reality could follow.

THE ARTISTIC COMMUNITY will see interactions that both address the developing practice of interactive art and inform sculptural practice. The creative tools show a new way of interacting with space, with tools that do not depend on mathematical language. Here there is a special relationship of body to digital form. The model of interface here gives a way to understand the interactive process that can be applied to many other types of interactive sculptures and displays with broad application in the arts.

part I:

context and approach

1 history and purpose

across the divide

THE WORLD AT PRESENT is faintly divided — material lies richly before us, infinitely malleable and manipulable at the human scale. I can rearrange the furniture in my room, shuffle cards, dress myself, and so forth, elaborately rearranging space as I do so. This is all done with ease and hardly a thought to technique, especially by those without physical handicap. Digital representations of information have also grown quite rich and manipulable, as addresses, pointers, links, other symbols are shuffled and re-arranged, labelled, collected, and integrated into wildly divergent digital forms. Visual information, 2d plots drawings, texts exist in multi-layered forms with varieties of links between them, and are moved and manipulated quickly with mice, joysticks, and other 2d input devices. But the world of three-dimensional data, whose structure so closely mimics the material with which our levels of fluidity and dexterity are at the highest, suffers at the hands of conventional input devices. While humans have managed to create 3d structures that command respect, the methodology to produce these artifacts is stubborn, brittle, and unwieldy. Moving a 3d point, a simple task in natural space, typically involves several separate motions on different 2d planes, separated by a slew of camera motions and slight adjustments.

The purpose of this text is to close the gap between the fluidity of physical, material manipulation and the (at present) unwieldy digital manipulation of 3d data. The goal is to have the fluidity, the essence of ease that physical law provides without limiting design to a mimicry of nature. Effective 3d interface techniques find sweet spots in spatial logic where digital manipulation is easy, because in this space we can set our own rules, ignoring gravity and rapidly transmuting our tool set as desired. This works offers improvements to interaction, in part accepting and leveraging physical intuition, in part extending control with a magical richness enabled by the great power of software to author experience.

The focus of this thesis is **spatial construction**, which includes both creating forms *ab initio* and rearranging existing data to make new structures. This area makes heavy use of the spatial manipulation that is so easy in the natural world, leveraging mechanical metaphors and cultural history to facilitate interaction. Much of this exploration is by example, through the analysis of interfaces that allow the manipulation of digital material. Digital surfaces are traced in space by the hand. Sensed tongs grab, pull, and stretch shapes. A handle tool places objects, and a raygun is used for pointing and selecting operations. The discussion of these tools occurs both in the abstract, through a classification of different modes of interactions and how they are coupled with physical tools, and in the specific. In the latter case examples show how these interfaces are used to create freeform geometry and manipulate molecular structures.

While these interactions, as the user studies throughout the text indicate, have immediate application, the further goal of this text is to understand 3d interfaces and nurture their development in the future. This work fits cleanly on a historical trajectory of increasingly rich graphical representation, tangibility, and immersive experience that has grown alongside the maturation of computers.

METHODS OF MANIPULATING SPATIAL DATA: A HISTORY

Humans can naturally directly manipulate material, it is only recently (since the Industrial Revolution introduced machines and techniques of mass production) that a more abstract degree of control has been found. Computers walk this history in reverse. The early incapability of machines to provide direct, intuitive, primary control was in part due to the limitations of computer input devices and displays, but even more due to the limitations of processors and memory to handle the (at one time) vast amount of data required to sense, process, and display 3d data. Around the early 1990's, computers caught up to our bodies, and it was no longer so difficult to interact directly with 3d data. Despite some rather successful experimental interfaces, immersive 3d modeling has not become commonplace. It is in part because a new understanding of interface is necessary to make these systems fully effective for 3d spatial construction. It is the lineage of tangible interaction, in full form only in the mid 1990s, that this thesis borrows from to further the effectiveness of 3d construction.

We now consider some transitions in the history of interaction with data, which range from very simple interactions to sophisticated systems for managing quite rich spatial forms.

Computers before realtime interaction

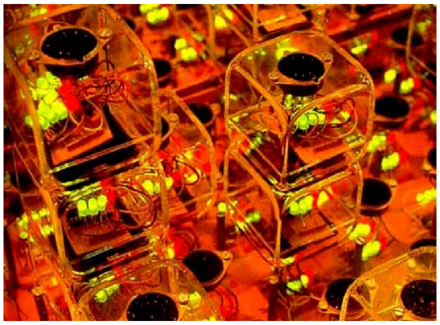
The earliest computers, such as the BRLESC-II system shown at right, were almost exclusively symbolic. Special machines translated numerical data into physical cards that were then input into computers. Before Von Neumann architectures grew prevalent, users set the current program of a computer by changing its configuration. In this time (the 1940s, 1950s, and 1960s) data was manipulated via custom displays which required sophisticated knowledge of the internal workings of the machine.

Early spatial construction

One of the first systems with an interactive display was the TX-II, built at the MIT Lincoln Lab. This system was designed with a light pen that could directly affect images on the CRT. Perhaps the first system to allow users to interact directly with graphical elements, Ivan Sutherland's Sketchpad was actually quite symbolically focused. In his PhD thesis [Sutherland 1963], Sutherland describes algorithms for solving for constraints drawn with the light pen. Using a pen to directly paint in a material fashion did not occur until much later, mainly due to the limitations of computer graphics hardware. Early 2d painting systems, such as Shoup's SUPERPAINT, did not emerge until framebuffers appeared in the early 1970's [Shoup 1975, Smith 1997].

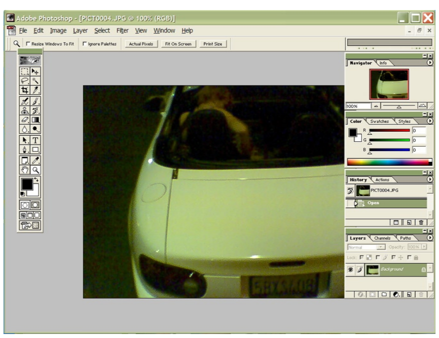
Early video games

Video games have traditionally presented some of the strongest links between the body and the motion of a digital character on a screen. While games emerged as early as 1958 [Brookhaven 2003], the first commercial system, Ralph Baer's Magnavox Odyssey, did not appear until the early 1970s [Winter 2003]. Video games depend on user presence and engagement, they are a paradigm of effective embodiment of humans in electronic space. Most video games have simple controls, such as a joystick, that directly map to the motion of an onscreen avatar. The early video game Pong, shown here, works in this fashion — players control the image of a rudimentary tennis racket.



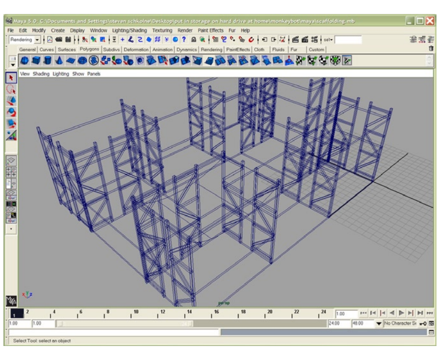
Sensing physical devices

In the early 1980's, the Frazers began researching experimental interfaces which directly link physical blocks to a digital representation. Sensors inside the blocks (as shown at left) allow their configuration to be sensed by a computer program [Frazer 1980, Frazer 1994]. This work was largely ignored by the research community until some fifteen years later, when interfaces such as the Triangle system [Gorbet 1998] emerged. Triangles are modular units which link with one another; sensors transmit the topology (but not the geometry) of the triangle grid to a personal computer. The chief limitation of this approach is that, while a physical structure can inform a digital space, it is quite difficult for the digital representation to affect the physical model. Moreover, the models are much less versatile than that offered by a digital representation. Anderson, Frankel, and others developed a system [Anderson 2000] that displays a stylized rendition of the physical model on a 2d display. This enriches the design space somewhat, without addressing the fundamental inability (at present) to digitally control physical form.



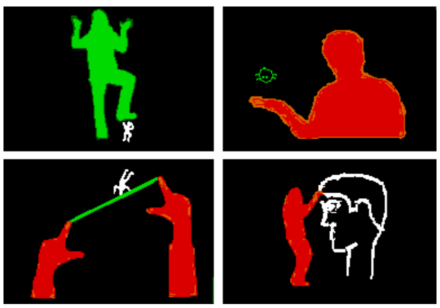
2d digital design

The early 1990s saw the emergence of a new lineage of computer applications: two-dimensional design tools such as Adobe Photoshop, Fractal Painter, Quark XPress, and Adobe Illustrator allow a new sophistication in the manipulation of static images. Also developing at this time is a new line of tools for the creation of dynamic and interactive content, beginning with Apple HyperCard and continuing through tools such as Macromedia's Director and Flash. Images from these tools populate magazines, galleries, a brief CD-ROM history, and the internet. These applications are not only rich in content, but moreover have low barriers to their use — no-budget publication efforts are almost defined by the use of these tools.



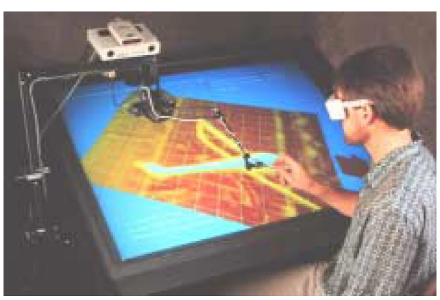
3d digital design

In the early 1990s, interactive 3d modeling became widely available, gaining both sophistication in use and prominence in the creative process for mechanical engineers and industrial designers. Alias PowerStudio, CATIA, and proprietary CAD systems are used to create cars, buildings, machines, everyday things. Coupled with animation tools such as SoftImage, and Alias|Wavefront's Maya, these tools revolutionize the production of cinema, both in animation and live-action where tools such as Discreet's Flame provide filmmakers the ability to create rich hybrid realities. Unlike 2d production software, these tools have high barriers to their use — until recently they required expensive workstations, and still require extensive system-specific training before they can be used effectively.



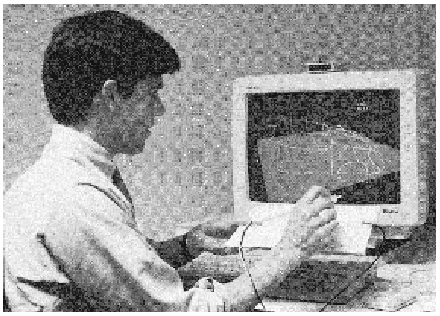
Conceptual approaches to immersion

Myron Krueger's VIDEOPLACE, shown here in its 1991 form, establishes a direct relationship between the entire body and digital space [Krueger 1983, 1985, 1990]. In this system, the user's silhouette becomes a character in a wall-size two-dimensional display. As early as 1969, Krueger and Dan Sandin were developing a concept of physical coexistence of humans and data via whole-body input that Krueger describes as Artificial Reality [Krueger 2000]. In the early 1990's this concept was popularized as Virtual Reality — a concept focused more on 3d interactions with a sensed glove and head-mounted display, whose grail is the total immersion of a human in a digital space. Augmented Reality focuses on the realtime compositing of data onto the physical environment, typically through a head-worn see-through display [Feiner 1993, 1997]. Another approach to Augmented Reality is to project data directly onto the environment using projectors [Bajura 1992].



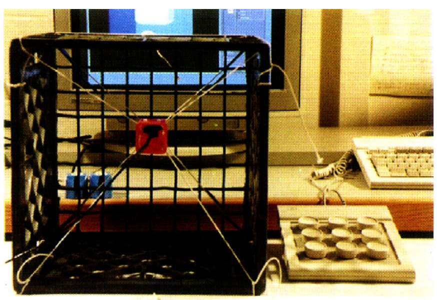
Displaying forces

Haptic devices, which display forces to users, originated with early prototypes at the University of North Carolina [Batter 1972]. Researchers have almost exclusively used robotic arms to transmit forces. These devices have gotten progressively smaller and lighter, such as the one shown at left that is used by the Nanomanipulator project [Taylor 1993]. Another approach to displaying the sense of touch is to control the height of a thin surface. This is known as tactile display [Kawai 1996].



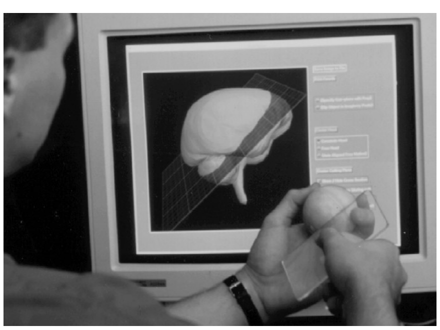
3d spatial construction

Perhaps the first system to allow the direct input of 3d spatial data, 3-Draw [Sachs 1991] uses a 3d stylus to place the control points of 3d surfaces in space. Much like Sketchpad, 3-Draw is primarily concerned with placing points and constraints in space. The HoloSketch interface [Deering 1995] uses a stylus to trace curves in a stereoscopic monitor-size display. This system is focused on a more material mode of direct creation. Pie menus are used to change the stylus's function, from tracing curves to placing primitives to simple animation controls. These two projects are indicative of a general area of research into using a stylus with a monitor-sized display to create 3d objects, such as JDCAD [Liang 1993], and Shaw and Green's two-handed polygonal modeler [Shaw 1994]. Wesche and Seidel present more recent results that use the Responsive Workbench as a display [Wesche 2001].



Computational clay

Applying haptic display to mimic the experience of clay sculpture is an area of active research [McDonnell 2001]. The early work of Galyean and Hughes [Galyean 1991] includes an experimental haptic input device, seen at left. Unlike conventional haptics, this mechanism is not attached to the user and provides a limited amount of feedback representing the local resistance of a surface. Unlike the work in surface and curve generation, much research in this area has focused on computation, as volumes are inherently more data-heavy than surfaces [Wang 1995].

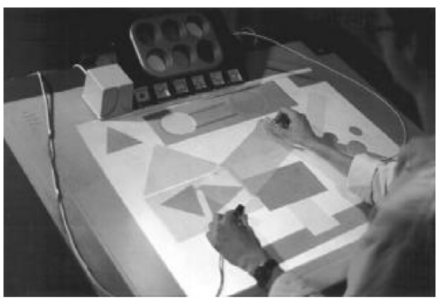


3d props and two-handed interaction

Hinckley, Pausch, Goble and Kassel's interface for neurosurgical visualization is perhaps the earliest example of the use of 3d props as input devices [Hinckley 1994, 1997]. Hinckley's props are an early example of cultural affordance, where the design of an input device is significant because of its everyday presence outside of computing environments.

Diversification of virtual reality

The early 1990's saw several new approaches to virtual reality, such as the CAVE — a room-size display [Cruz-Neira 1992] and the Responsive Workbench [Krüger 1994], seen at left, which presents a 3d space above a table's surface. Alongside this work are several methods of navigating virtual space – ideas range from letting users manipulate a small version of the world [Stoakley 1995] to redirecting the walking of users [Razzaque 2001] to changing the relationship of body to the 3d space [Pierce 1997 & 1999, Poupyrev 1997].



Tangible user interface

In the Bricks interface [Fitzmaurice 1995], physical bricks are used to group data elements that live on a desk's surface. In the late 1990's, several projects at the MIT Media Lab explore this concept further, notably the metaDESK [Ullmer 1997], and the mediaBlocks system [Ullmer 1998], which further investigate the union of functionality and physical identity. The Luminous Room [Underkoffler 1999] explores the combination of 3d passive components and 2d projected display. The Block Jam interface [Newton-Dunn 2002] allows the control of music with a set of tangible blocks.

CONTINUING THE TRAJECTORY

The remainder of this text investigates how immersive displays and tangible input devices can aid the spatial construction process. In the spirit of Hinckley's two-handed interfaces which establish reference frames between body parts [Hinckley 1997], the new tangible input devices are designed to establish a strong link between body and space that enhances interaction. Cultural affordance, kinesthetic framing, and the direct union inherent to immersive display are the means to this end.

Spatial construction methods are typically either good at artistic design or engineering design. The most successful designs incorporate elements from both avenues of thinking. Two major application areas: freeform surface design, and DNA modeling, explore these opposite poles of the modeling process.

The larger goal of spatial construction interface is to unite the mind of humans with that of their designs. As this brief history shows, from the 1960's to the late 1980's, the sophistication of models increased alongside developments in framebuffers and geometric rendering with little change in input devices. In the past 10–15 years of research, advances in display have been outstripped by those of input devices. Input devices are finding a new maturity, through 3d sensing and a new understanding of affordance, and it is these two advantages which are employed to allow humans to fluidly create, deeply embedded in their design space.

2 contributions

philosophy, practical experiments, theory, and validation

Q: *What new territory does this thesis explore?*

A: This thesis explores the intersection of tangible interface and immersive 3d display, presenting an approach to 3d spatial construction and insight about interface at large. The contributions are

- practical components — new tangible tools, new interactions based on those tools, and new interactions for existing tools

- theoretical components — concepts of kinesthetic framing, and direct union which explain the effectiveness of spatial interaction, an analysis of cultural affordance as a familiarizing factor that makes interface more comfortable, and an attention to the mapping between input and action parameter spaces embodied by both direct union and the relationship between discrete and continuous variables

- insight — numerous examples show how 3d interface is best when it is based on coordinate frames instead of point selection, and how multiple accessories can be controlled by one physical base to support a large range of functionality with a small toolset.

- a philosophy — viewing a tool's form as embodying a class of functionality informs design, and allows a strong support, both culturally and kinesthetically, for actions in digital space. This approach strikes a nice balance between generic and specific interface.

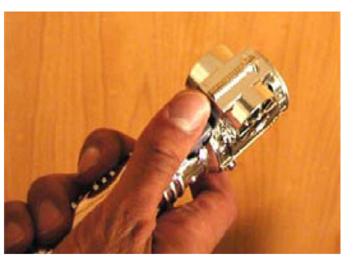
Short-term user studies and prolonged collaborations with artists and scientists demonstrate the effectiveness of these techniques.

Q: *So, for starters, what are these new tangible tools?*

A: Actually none of the tools are entirely new — they all have a history dating back centuries. The tongs and raygun are new to use as 3d input devices, and all four of the devices below are used in new ways.

The tongs move, scale and rotate objects at several levels of detail.

The raygun selects points in space, its function is the closest analogue to the mouse in the 3d setting.



The handle drags digital tools through space, tracing curves, and placing objects

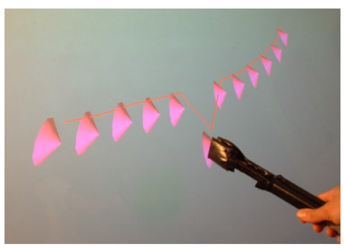
The hand traces shapes and subtly deforms space. The hand's realtime control of curvature and orientation is its chief benefit here.

Q: *Why did you pick these tools? What can I do with them?*

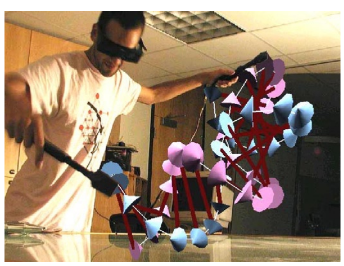
A: These tools were chosen because they span a wide range of relationships to space, both in the culture of their use and in the way the body relates to space when they are held in the hand. Here are some examples of the interactions that are supported by these tools.

Tongs:

The tongs move shapes. While it is somewhat trivial to use the tongs to move an entire scene, moving one of many shapes, or a small piece of a larger shape requires specialized phrasing.

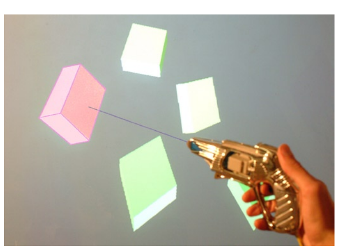


This thesis develops a hierarchy of Reframe actions as a model for different types of motion that can be performed by the tongs, such as moving a single molecule in a chain of molecules, as shown at left.

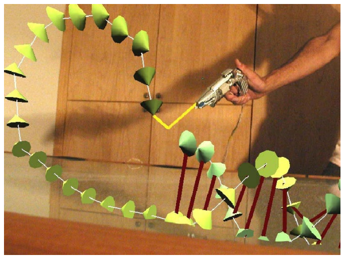


A two-tonged interaction stretches shapes. The tongs are an effective component of many two-handed interactions, often establishing a rough coordinate frame for more precise operations with the dominant hand.

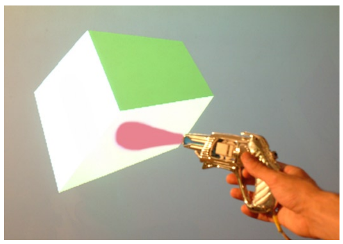
Raygun:



The raygun selects objects in space. Much like the 2d mouse, the raygun is well suited to point selection.

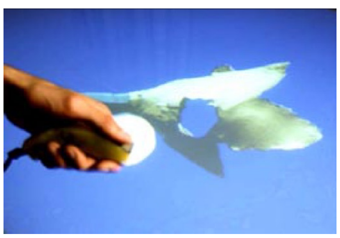


The raygun also draws links between objects — in this case, bonds between DNA base pairs.

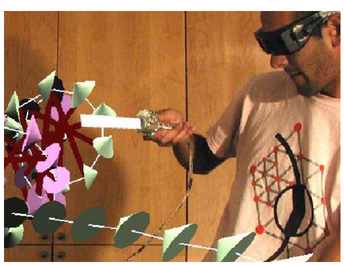


The raygun is best used to select points or a range of points on an object. This type of action can be used in different ways, such as spraypainting a surface.

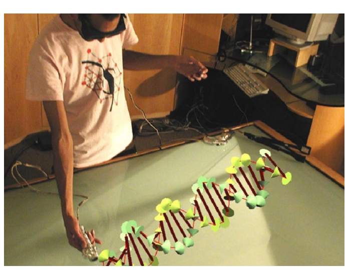
Handle:



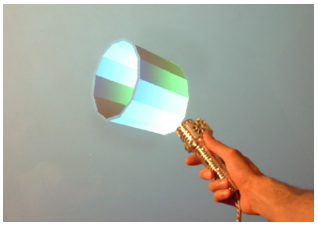
The handle can hold a variety of tools, such as this eraser which removes a small volume from an object.



Here the handle controls a lightsaber which cuts a bond between DNA bases.

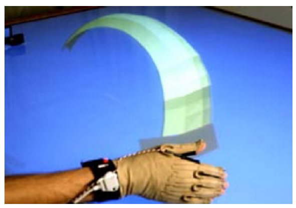


The handle's actions are not limited to moving tools through space, it can also be used to draw freeform curves or more complex objects such as this DNA double helix.



The handle can also place objects. Here a cylinder is created with the handle. This tool allows the cylinder to be placed with a specific orientation.

Hand:



The hand is used to trace paths in space. These strokes can be combined to form organic geometric forms.

The hand can also smooth and deform objects. The image at left shows the result of deforming a flat stroke with the hand.

Q: *If I wanted to unify the themes you have explored, and extend them and apply them to other domains, how would I do so?*

A: Each tool represents a class of functionality. As the tongs are a generic grabbing tool, so the raygun is a kind of generic pointing tool. As you have seen above, each type of tool can be applied to several sets of tasks. If you want to extend this system, you might use the existing tools and map new functions to them. If you are doing a similar spatial construction task, this might be appropriate. For a very different type of application, you might have some new tangible tools, but the same basic idea would apply.

Q: *Is this what you meant by your philosophy above?*

A: Yes, when I talk about a class of functionality, I'm referring to groups of actions like grabbing, tracing, holding, and pointing. One of the unique things about this work is that it takes each class of functionality, identifies a type of 3d tool that supports this class, and proceeds to show how it can be used for many applications. Prior work on tangible interfaces tends to map input devices directly to data, not to functionality.

Q: *If I wanted to design a new class of tools, what should I pay attention to?*

A: There are two major points to think about, both of which involve terms I introduce, define, and elaborate upon in this text. The first is **cultural affordance**. You should think about what devices are used for your target task outside of the computer, and try to imagine sensing them and mapping them to digital operations. One theory of this thesis is that these cultural affordances provide a deep intuition, immediacy, and fluidity of interaction that enhances a user's connection to digital space.

The second concern is what I call **kinesthetic framing**. You should think about this whether you are building a new tool, or simply extending the functionality of the existing tools. This concept refers to the spatial relationship between the body, device, and the function that the tool form defines (and proceeds to mediate as it is used). Appropriate kinesthetics facilitates physical action and provides a cognitive clarity in a user's mental model of space.

Q: *These interactions are all interesting, but what can I use them for? How far do you go beyond theory into real applications?*

A: Two application areas are focused upon: freeform surface creation and molecular design. User studies of these systems, in addition to a small user study that focuses upon a common rearrangement task, show how novice users respond to these systems. Additionally, I have worked over considerable periods of time with groups of artists, and report the results of these collaborations.

Q: *So what makes these techniques effective?*

A: The key to these interactions is the way the input devices frame digital space, subtly guiding the body and mind. The directness of this link provides an immediacy which has emotional benefits, and the ability to input large amounts of data rather quickly. This text lays the foundation for further development of these methods, explaining the critical factors for successful interaction in this setting, and providing techniques for providing a wide range of interactions with a few versatile, appropriate-to-the-task, 3d input devices.

3 spatial construction

the range of tasks which will receive our focus

SPATIAL CONSTRUCTION is defined by its result: some artifact which lies in space, something that coordinates can be assigned to. At times these coordinates have additional nonspatial information such as material properties. Techniques for achieving this result have a deep history rooted in carving tools, assembling materials to build shelter, weaving, and making clay bowls. In the current age of mass production, the vast majority of spatial construction is performed by specialized groups of people with detailed plans. Cars, furniture, appliances, and buildings are typically designed with a preliminary digital representation to guide the process. Some less typical but no less relevant spatial construction tasks are circuit layout, painting, and acoustic design. Any task whose end product is an object that lives in space is within this domain.

Tools for spatial design are part of early education — blocks, clay, scissors and glue, all of these are familiar from childhood (for the American reader, at least). Industrial designers (with the help of a few power tools) still use similar methods to make prototypes (although the practice is being replaced by rapid prototyping from digital models). Spatial interaction forms a foundation for thought. People genuinely think in spatial terms; if you ask an athlete how they hit a baseball they will have no language to explain it, while this mind is in fact performing complex control computations. The tight rela-

Tinkertoys are a children's toy that allows them to play with ideas of structure in a very physical way. Such toys introduce and reinforce concepts critical to spatial analysis.

tionship between thinking, vision, and form is elucidated by Rudolf Arnheim's seminal text Visual Thinking [Arnheim 1969], and further illustrated by McKim's illustrated book Thinking Visually [McKim 1980]. Readers are encouraged to consult these texts for background on this important aspect of cognition.

Recently digital representations have become the standard model for spatial objects prior to fabrication because they exhibit an ease of duplication, an unbounded precision with which elements can be represented, layers upon layers of complexity (a building can be represented down to wires, nuts, and bolts), an ability to quickly change aspects of a design such as material properties, and the nascent ability to predictably simulate the behavior of the spatial object in a variety of lighting conditions, or under the impact of forces and strains.

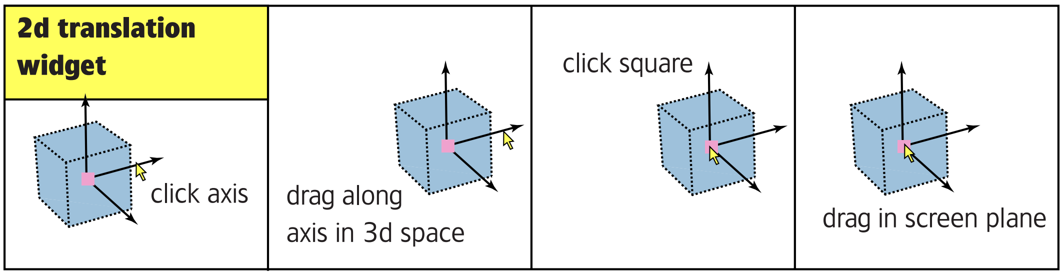
Digital representations are powerful, especially for groups of people working in concert. Yet the software interfaces through which these representations are manipulated are more difficult than physical construction. While digital artifacts are inherently more complex than physical ones, we note for even the simple physical task of clay modeling, computer systems are not as powerful as the physical clay.

Our aim is to find a physically, spatially intuitive way to interact with digital spatial representations in all their complexity. The remainder of this text develops a framework for understanding 3d spatial construction, and several interactions that have proven effective at supporting this common, important practice in the digital realm with full efficacy.

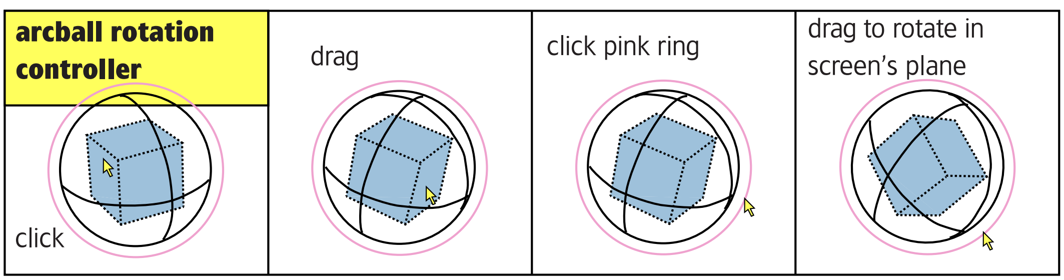
THE SUCCESSES AND SHORTCOMING OF 2D INPUT

Using 2d mice to create and manipulate complex 3d structures is ludicrous, absurd, on some intuitive level the mapping cannot be direct enough to be functional (pause for a second to think about this). Other 2d input devices such as trackballs and pens suffer from the same inherent dimensional mismatch. Yet humans are inventive beings, they have not only created software that effectively uses the mouse to create shapes, these tools have become the de facto standard for motion picture creation, the aerospace industry, and many other domains.

Many spatial construction actions can be accomplished with a 2d mouse with 3d widgets that translate 2d motion into 3d space. Navigation is often performed with a set of operations, each mapped to a 2d mouse button. For example, pressing the left mouse button and moving the mouse translates a scene. The middle mouse button triggers rotations and the right one triggers a scale. Throughout this style of interface there are ways to configure the center of rotation, and ways to specify whether the translation is in the screen plane or in some other plane. For translation, a popular approach is to place a 3d widget (a coordinate frame with x, y, and z axes on it) to guide motion along specific directions.



Mapping 2d inputs to 3d manipulations suffers an inherent bandwidth limitation. Users have to perform several inputs for a single action — moving a cube with the 2d translation widget shown above requires several clicks and drags. Humans are accustomed to moving and rotating objects at the same time with their hands or handheld tools. With 2d input devices, rotation is similarly split into several operations. Consider the arcball rotation controller, a popular method for rotating objects with 2d interfaces:



Picking is somewhat simpler with a mouse, in sparse scenes a user just has to click on an object to choose it. In denser data sets, seen in molecular visualization applications, clicking a component atom can be quite difficult due to visual density, and users must navigate until the component is in plain view before selecting. Users sometimes select structures by typing in their name. In 3d modeling, sometimes a scene graph (displayed in a separate window) is used for this purpose. These last two interactions punt on letting users operate spatially and instead allow them to work directly with the information structures underneath the visual model. While this style of interaction is the most appropriate for some tasks, in this text we focus on spatial construction where the user desires a direct manipulation of spatial properties.

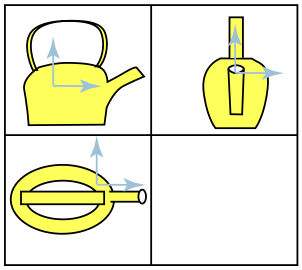
With 2d interfaces, objects are placed with a click. In some cases objects are placed in a default location, in other cases objects are placed at a position related to that of the mouse click. These interfaces do not allow full control of placement. At this point we begin to see a trend in the use of mice for 3d input: a simple action is surrounded by further manipulations which complete the user's intention. When placing an object, the simple action of clicking to introduce it is necessarily followed by a slew of motions and rotations which place it precisely in the scene. In the picking example above, the simple action of clicking an object to pick it is preceded in time with a series of navigations required to give access to the particular object. Interfaces such as SKETCH [Zeleznik 1996] merge these two actions by allowing the user to draw the coordinate frames of the objects as they introduce them. This approach has some drawbacks, namely that for complex objects such a coordinate frame is not implicit — in practice SKETCH uses a few styles of drawing coordinate frames for each of a few objects that can be placed: cylinders, extended cubes, and curves. Perhaps the richest interpretation of 2d input is found in the work of Igarashi [Igarashi 1999 & 2002], where freeform models and cloth can be controlled with specific mouse motions.

MICE ARE POINT INPUT DEVICES

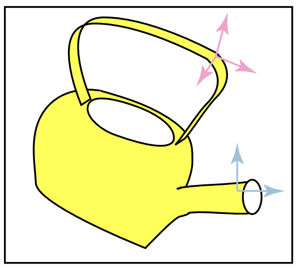
Operations such as sweeping surfaces, erasing volumetrically, and creating volumes are difficult to directly map to a mouse. In practice these are achieved in mouse-based systems with a additional internal widgets such as sets of curves, planes of deformation, and CSG operations.

In the early days of interactive computing, data storage was limited and representing 3d structure as material (a large array of volume elements or a dense polygonal mesh) was difficult. The earliest representations described continuous surface with small sets of numbers: spline surfaces [de Casteljau 1959, Bézier 1970]. Spline basis functions can control a 3d patch with as few as four 3d points (more points are often added to increase control). These patches can be tiled to make a larger surface. Since these patches are specified only via points (sometimes orientation is also used, accessed via points on the tangent plane) they are easily mapped to a mouse and orthogonal views.

Equipped with mice, users interact with 3d data in two fundamental ways:



Three orthographic 2d views provide separate images of the side, top, and front, of an object. The mouse is used to move points in the screen plane (blue arrows).



Using a perspective view and specialized metaphors, users move points either in the screen plane (blue arrows) or along canonical directions in world space (pink arrows).

With this ability to place and move points, and the mathematical nature of spline surfaces, users can manipulate symbolic forms to create a marvelous array of freeform shapes.

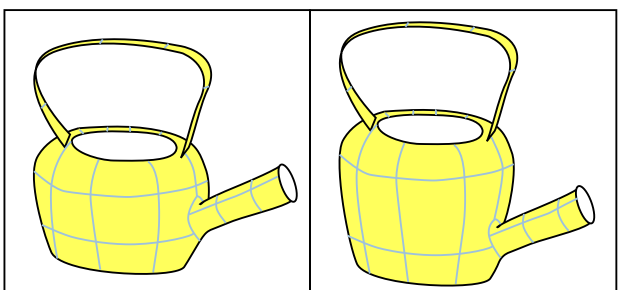
SYMBOLIC VS MATERIAL METHODS OF CONSTRUCTION

The manipulation of curves underlying surfaces is an example of **symbolic** manipulation. Here an underlying construct affects the geometric placement of shapes. We can think of this class mathematically: the final shape is a function of an underlying control structure. Affecting this structure indirectly changes a user's design. This approach contrasts with the **material** manipulation seen in traditional artistic methods such as stone sculpture. Users directly control geometric form in material construction methods. Each view of construction has its own set of advantages. Material manipulation is more friendly for direct operations for which a notion of structure is not necessary. For example, placing a bump on a surface with virtual-clay style interaction [Galyean 1991, Wang 1995] is easy, whereas with a spline surface this can be difficult if the symbolic structure does not afford the appropriate control.

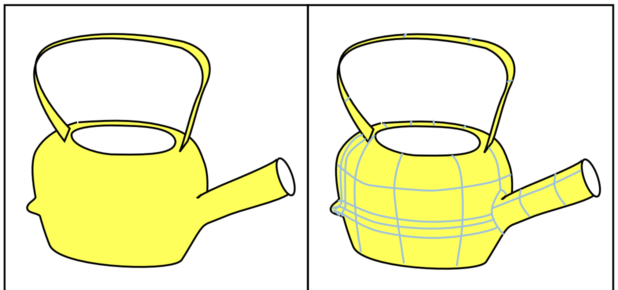
Clay interaction is perhaps the epitome of material construction. With clay, a homogeneous material is manipulated with the hands to achieve a certain form. This method has many advantages, including the deep presence of clay throughout history, the degree to which artists are trained in this method, and the straightforward manner of construction which it admits. Clay, and similar methods, have an important and continuing role in modeling that computers will not claim for many years. This text stays away from the metaphors of clay, not because they lack value, but rather because computers are best when their unique strengths are maximized. Digital spatial construction is not unlike digital music — it has taken many years for the properties of natural instruments to be mapped to digital space, and in the interim a rich world of purely digital music has come to maturity. It is this type of new territory which digital modeling offers, not a replacement for successful traditional media.

The chief advantages of modeling in digital space are akin to the strengths of binary representations which have caused the proliferation of computers throughout society. One can change the lighting properties of a model, or test it in various environments, with much less effort than it takes to re-paint a clay form. Digital structures can be easily duplicated, the representations are highly abstract, both allowing the structured modeling methods which are discussed next, and moreover allowing us to bend the rules of gravity. In digital media users can work without scaffolding and physical support for their models. While physical frames are not necessary, often mental scaffolds are built, and these symbolic structures are both a strength (the user can make global changes quite easily) and a limitation (users can not make simple changes if they have built an inappropriate scaffolding).

Here we see the strengths and weaknesses of digital symbolic control:



Structural elements, such as the spline control network shown in blue, are particularly useful for large-scale deformations such as the action of stretching the teapot to a new height.



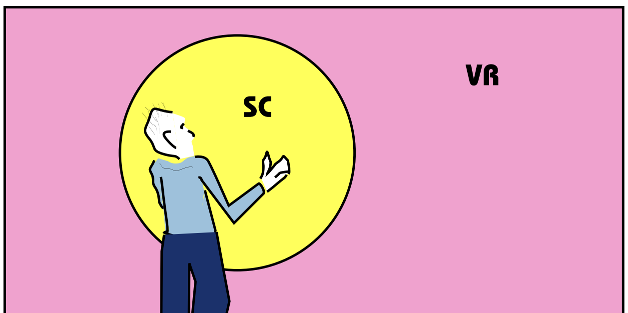
Small changes such as the bump at the back of this teapot are easy to incorporate into material methods of construction where the surface is directly manipulated. Achieving the same effect with the original spline structure (seen at right) is quite difficult, requiring several new control elements which complicate the structure elsewhere in the design. As such, the initial design of a symbolic support can be quite important.

One of the chief benefits of symbolic modeling is its use in animation. Virtually all digital animators use high-level controls for their art, and often there is a strong correlation between the symbolic structure and animation controls. For digital puppetry, animators build character rigs which are tuned to express a desired range of character motion. These structures need not be present during modeling — character controls can be added to unstructured meshes after modeling is complete. The controls also are sometimes affected visually, and sometimes numerically. There are many conventional 2d interfaces for manipulating control structures for animation. In some production settings, such as that at Pixar where a very fine degree of control is needed, animators explore parameter space directly, with a spreadsheet-style interface for sending a large number of options to an offline renderer [Meyer 2003].

Many symbolic operators are seen in the InDex modeling system, a descendent of the innovative SmartScene system which was shown in SIGGRAPH 1997's Emerging Technologies exhibit. These products, currently developed by Digital ArtForms, allow users to place objects and navigate space using generic controllers in a 3d space. InDex supports an industry-standard CAD back-end, allowing curves to be placed and swept to make surfaces. The chief difference between InDex and the interfaces developed in this text are that InDex uses highly generic input, the form of which is weakly mapped to the task at hand.

SCALE RELATIONSHIPS IN SPATIAL CONSTRUCTION

Spatial designs come in many sizes, from the microscopic to the gigantic. We will see a wide range in this text, from a molecular modeling application to a freeform drawing application whose range includes architectural scale. Scale is an important consideration, in particular the 3d interfaces occur at a bodily scale. Data need not live far from the body; this is one key difference between spatial construction and virtual reality.

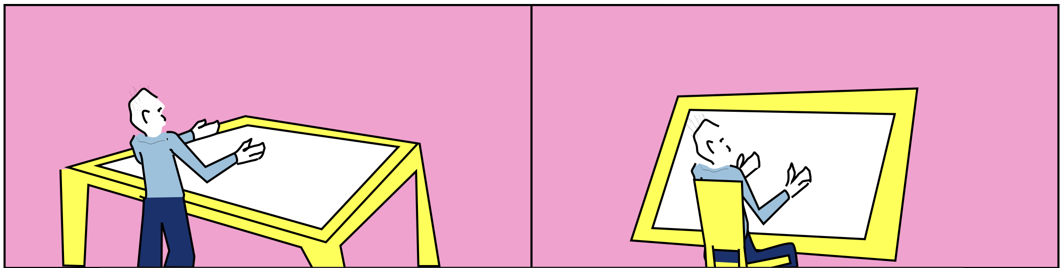


The active spaces of 3d spatial construction (SC, yellow) and virtual reality (VR, pink)

In spatial construction, it is ideal to have most of the data within range of the body, where the most sophisticated 3d manipulations can be performed. Virtual reality concerns itself with presenting large-scale environments that completely immerse the viewer. In this sense, the two domains are fundamentally different.

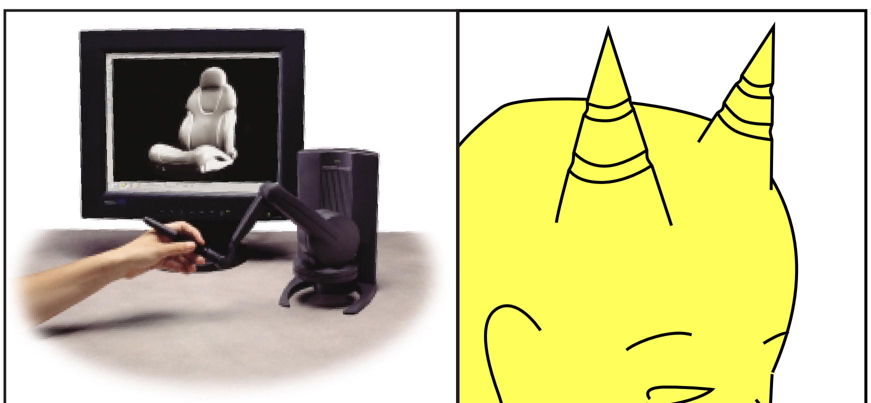
Much of the early work on immersive modeling worked on monitors which restrict the body's motion to a smaller screen. In these works we find an important early precedent for symbolic modeling [Sachs 1991]. Of particular interest is HoloSketech [Deering 1995], in which curves are directly traced in space in front of a stereoscopic monitor. Some systems use a larger display, such as BLUI [Brody 1999] and CavePainting [Keefe 2001]. In these setups, users still tend to work close to their body — in CavePainting the model tends to live between the wall of the CAVE and the user's body.

The interfaces in this text were implemented on two types of display — the Responsive Workbench [Krüger 1994] and an immersive projection screen (both are shown below). These displays offer a lot of room for arm motion, but much less space for body motion than the CAVE presents.



The Responsive Workbench (left) and a body-sized flat display. In both cases stereoscopic imagery leaps into the space around a user's arms.

Restricting the scale of operations even further is a sacrifice the Freeform modeler makes to add haptic feedback. SensAble, the manufacturer of both Freeform and the Phantom device which serves as its input device, actually reduced the range of the Phantom device to improve haptic fidelity [SensAble 2003]. In comparison to immersive approaches, what is the utility of haptics? Deformations in this monoscopic system are not seem much easier than the haptic-free polishing interaction described in Chapter 6, as visual processing dominates this task. Haptics are most useful for tasks which involve following lines of curvature on a surface, such as carving rings around a horn. Haptics is less important for moving objects, which SensAble controls with a mouse.



(left) SensAble's freeform modeling system uses a haptic arm, which is particularly useful for features such as the rings around the devil's horns (at right) which follow the curvature of the surface.

The Haptic Workstation offers a full haptic experience at the expense of quite a bit of user comfort.

The extreme of exoskeleton haptics is seen in Immersion's Haptic Workstation, which is not used to construct but rather to evaluate models generated in CATIA's mouse-based CAD modeler. This system places the user inside a large double armature to control their body as if they are interacting haptically with a 3d model. I used this system to feel a CATIA model of the inside of a car interior. Using this system impressed upon me both how hard it is to get the right sensation of haptic touch, and how useful the kinesthetics of operation were. The most interesting interaction was picking up a Coke can – not because it felt convincing to pick it up, but rather because it was terribly convincing to hold. A big difficulty with this Stelarcian [Stelarc] approach is the time it takes to get into the system, and the limited range of the body once fully encumbered/enabled by the I/O devices.

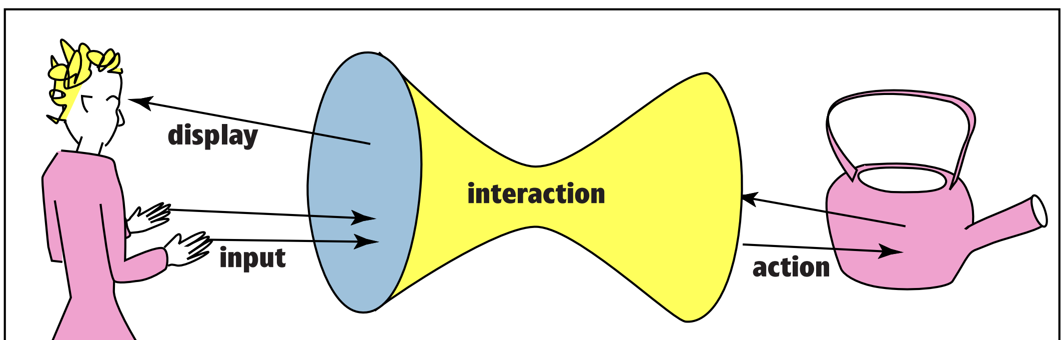
part II:

devices and interactions

4 conceptual model

the interplay of actions, inputs, and display

INTERFACE, as classically understood, is a two-way street. There is a human, and on the other side of the channel there is some space, a digital representation, and as the interface is enacted there is some interplay between the two. Newer interfaces such as ambient displays, multiplayer games, and collaborative interfaces link many humans to one digital representation. Conversely multitasking links one user to multiple, (conceptually) independent data stores. The double-funnel image below visualizes this relationship in the case where there is one user and one set of data. This model visually represents the interrelation of **display**, **input**, and **action**. Inputs are the physical motions of the user that activate sensors. In this text all input is performed by the hand (often with input devices). Voice and full-body input are widely studied elsewhere. Interaction matches inputs with **actions** which change the shape being created. The machine outputs information to the human's sensors (eyes, ears, hands, etc) via a display. In our case, although there is some output in the tactile feedback of tangible controllers display is largely visual. Interface design is the challenge of finding the right connections between input and action to make the channel represented by the double-funnel as fluid as possible. Design is difficult because the channel, the actions, inputs, and display must all work together. The double-funnel shows that there is an infinitude of options on both sides of the design space, and only a small set of sensible interactions connecting these vast seas of choice.



The double-funnel model of interface

Actions are internal changes in a computer's state which are triggered by input, and reflected in the display. In a mathematical sense, the system moves from state to state, and actions are operators which represent transitions in this state-space. Consider a word processor — the Character action (throughout the text the names of actions are capitalized) adds letters to the body text of the document, based on where the cursor is positioned. Clicking on the body text with a pointing device repositions the cursor (the Cursor-Reposition action), as does pressing an arrow key. Note that Cursor-Reposition can be defined without reference to either keyboard or mouse, and that it can equally well be mapped to eye gaze, or a voice command.

This combination of input, action, and display is at the heart of the interactive process, and as such should be at the center of an interface designer's mind. The fluidity of the interactions in an interface determines the effectiveness of a tool and the enjoyment of a user. Interactions are a gestalt: if an input is mapped to an inappropriate action, an otherwise effective input device can become unwieldy. If an interaction's display cues do not appropriately guide and inform a user, an interaction is unlikely to be of any utility whatsoever.

An important class of actions is what I will call **discrete actions**, which do not specify continuous properties. Discrete actions do not act on spatial properties (which are invariably continuous). The perfect example of such an action is Undo, and other operations such as Duplicate and Delete also fit into this category. They are important because they are indispensable for computer operation, but they are not emphasized in this text because most discrete actions are not particular to the act of spatial construction. In practice, conventional methods are used for these operations. An example of a discrete action is the Menu action seen in Chapter 7.

This text builds on the 3d widgets developed at Brown University in the early 1990's [Conner 1992]. This work was one of the first to focus on specific devices that could mediate a variety of tasks in 3d space. The 3d widgets are largely built upon the assumption that a user will be picking and controlling the position of points in 3d space. In the years since this work was completed, the importance of tangibility and kinesthetics in input has become clear, and hence this text looks at both the input device and the digital element which will allow for control.

An effective interaction is a special thing, the spaces of inputs and actions are both quite large, and connecting them well is vital to effective interface. Underneath physical action, the purer goal is to unite the mind of a user and that of the machine. Norman's gulfs of execution and evaluation [Norman 1986] treat the subject at a somewhat deeper, certainly more ambitious level than the input–action–display breakdown shown here. Instead of looking at input, Norman looks at the intention of the user, and similarly display is replaced by the understanding of the user. The design analysis of this text is physically rooted: it does not explicitly tread in the space of user intention and understanding, although it is implicitly a factor of great concern. Much of the justification for the designs found successful comes from user feedback, which constantly refers to the ease of interplay between mind and digital space.

5 arranging spatial elements

using tongs and a raygun to place, select, link, and reframe space

ARRANGING and placing objects in space is a large part of spatial construction tasks, even in the natural world. Imagine assembling a car's engine — one first has to get the components next to one another appropriately. In 2d document interfaces, much time is spent scrolling around the window. Similarly in 3d, users frequently place objects in space, bring them close to one another, bring the scene close to themselves, and move subcomponents of objects. If we were to look at a list of the actions that a user performs, we would see that arrangement actions are frequent, often surrounding other actions that change a model's state.

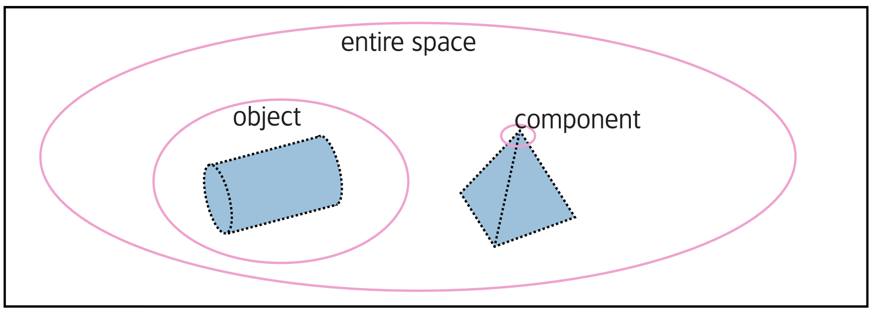
As seen in Chapter 3, 2d GUI interfaces to 3d space treat different components of 3d translation as separate actions. This system is particularly efficient when the desired action is represented by one of these actions. For example, moving a painting on a wall, when the coordinate frame of the widget matches that of the wall is quite easy. But placing a hat on the head of a character is difficult, not only because it requires integrated motion and rotation, but moreover because getting the rim of a hat to line up with the head requires an integrated control that is hard to represent with separated variables.

This issue is addressed by an early study of the integrality and separability of input devices [Jacob 1994]. Integral properties, such as translation and rotation, are perceived together while separable properties (color and translation) are not. It is desirable to control all integral properties in the same action. 3d interfaces ease the act of placing objects in space, both by providing integral control of the parameters in question, and moreover by establishing a direct union between objects and a user's body (more on direct union later in the text). Before delving deeper into the input devices used to manipulate and arrange objects in space, let's look at Reframe, an action which represents a wide range of positioning operations.

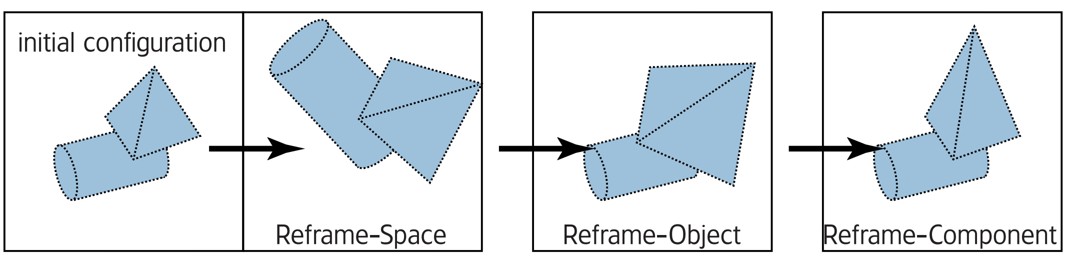
REFRAME: AN ACTION FOR ARRANGEMENT

In most computing environments it is only feasible to move the digital representation, not the user. Much research has been done in this area, the majority focusing on moving the user along a quite lengthy path in a virtual world [Stoakley 1995, Slater 1999, Razzaque 2001, Tan 2001]. In spatial construction the paths are shorter, often a single object just needs to be seen from different sides and at different scales.

Spatial construction manipulations are smaller in magnitude, densely clustered in one region of model space, and occur at different levels of resolution. Consider the scene shown below:



A user might want to change the shape and placement of a tiny portion of an object (a component), or alter the entire scene at once to change their placement within it (navigate). I call this family of actions Reframe, because in all cases the coordinate frames of the objects are being altered, and moreover in a non-mathematical sense their context, and the user's relation to them, is being re-staged. Reframe actions can affect the digital representation at each of these different levels of resolution.

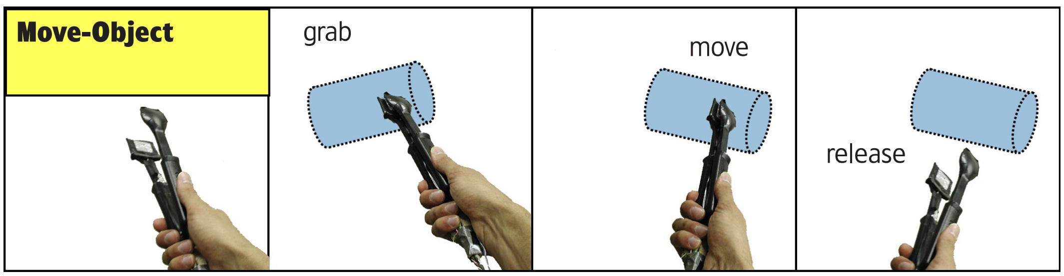


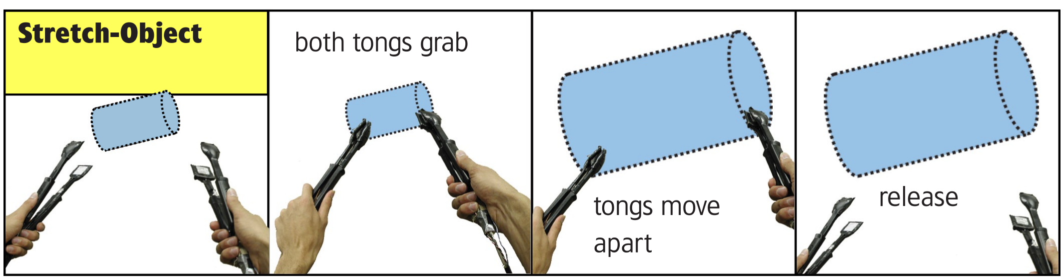
The above image shows three varieties of Reframe. In Reframe-Scene, the initial configuration seen at left is changed as a whole, both the cylinder and pyramid are moved and scaled. In Reframe-Object, only they pyramid is moved and scaled — the cylinder remains in its original configuration. In Reframe-Component, only one vertex of the pyramid is moved — the cylinder and the other pyramid vertices remain in the original configuration.

The Reframe action can be performed with similar input at each level of resolution (in every case the same type of data, a coordinate frame in 3d space, is being manipulated).

TONGS

Tongs are a tool for grabbing physical objects that (via sensors) can be altered to interact with virtual space. The act of closing them is a natural signal to begin a Reframe, for the duration of which the virtual object moves with the tongs themselves. Two sets of tongs used together can be used to stretch an object. Here are interaction cartoons which show how the inputs of the tongs can be mapped to these actions:





Note that it is quite nice to have two sets of tongs available, not only because Stretch can be implemented this way, but also because Reframe actions are frequent and it is nice to do them in parallel. Phrasing interaction so that one tong moves the scene while the other moves an object or component in that scene is non-trivial — what is to distinguish this from Stretch-Scene or stretching an individual object? Imagining this implemented with six sets of tongs borders on ludicrous. The seven (plus or minus two) objects that (according to cognitive psychology) can be kept in mind at once [Miller 1956] is a good upper bound for the number of physical devices a user can be expected to manage, although this bound is sometimes surpassed in traditional (non-computational) tangible interfaces (such as a mechanic's wrench set) where many physical devices perform unique functions. This issue is indicitave of a larger issue, particularly relevant in tangible interface: the distinction between generic and specific input devices.

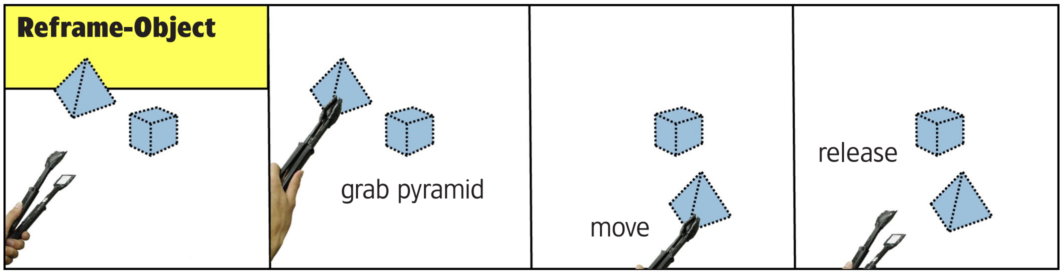
specific	**generic**
space-multiplexed	time-multiplexed
many phicons	few phicons
strong physical identity	weak physical identity

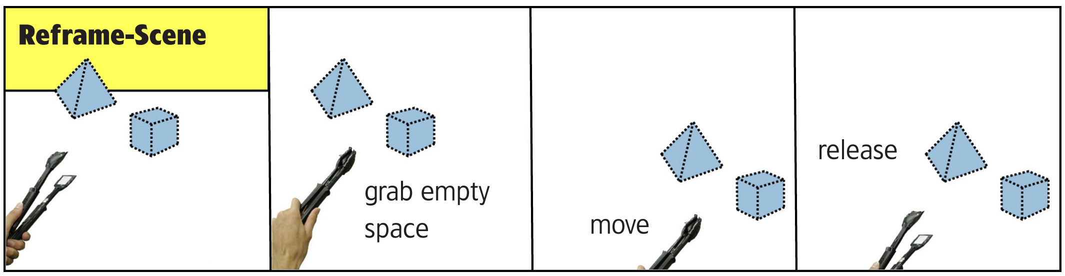
Mice are very generic, they are mapped to a vast variety of digital effectors across a range of GUI applications. Tangible interfaces such as Bricks [Fitzmaurice 1995] are also generic, although note that multiple bricks allow for a degree of spatial multiplexing even with this range of devices. More custom applications, in particular Underkoffler's Luminous Room [Underkoffler 1999] have custom devices which only serve one function in an application. In this type of highly specific interface, the digital representation of a tool merges with it's physical identity. There is something nice about interfaces that are so tailored to a task, although highly specific interfaces cannot take advantage of the high plasticity of computer systems, wherein rapidly changing software dramatically alters functionality without the delays and costs inherent in physical manufacturing, and without necessitating physical transfer of functional devices (software can be quickly updated via the net).

The tongs are somewhat specific — their affordance specifically shows that they will be used to move, stretch, and perhaps twist objects (the Twister algorithm is an ideal match for the tongs [Llamas 2003]). In natural settings it is clear which object a tong acts on, but the tongs are not as rich: they do not grab or twist all of space, and they cannot stretch most objects. One digital phrasing is to have the tongs grab the nearest object. Below a simple metaphor for the tongs is described. A refinement of this approach is seen in Chapter 7.

ONE TONG, MANY REFRAME ACTIONS

Consider a metaphor: when the tongs are close to an object, they grab it and perform a Reframe-Object action. When no object is near, the tongs grab empty space (Reframe-Scene).

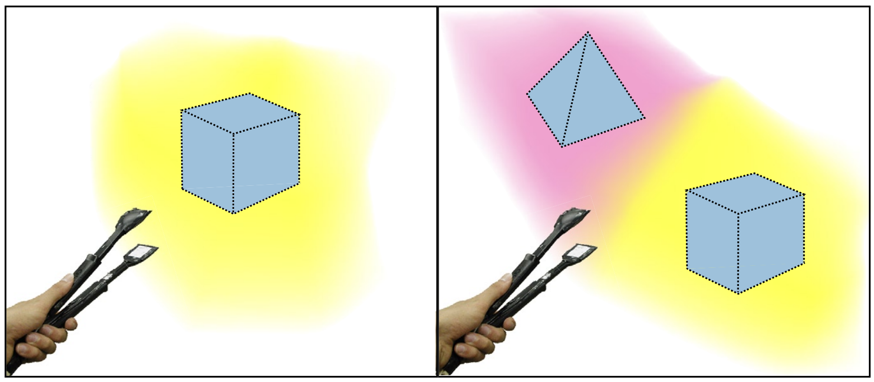




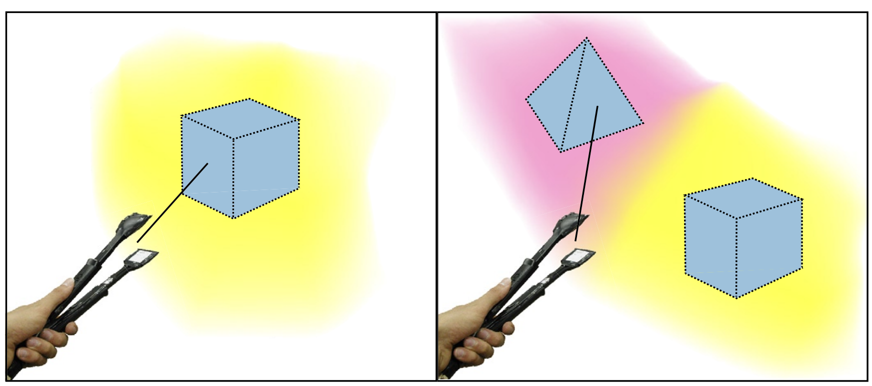
This strategy mimics natural experience, it sounds good in theory yet difficulties arise in practice. The depth cues are not as strong in synthetic space as in natural space. It can be difficult to grab specific objects in a crowded scene, and even harder to find empty space to trigger Reframe-Scene. This illustrates an important property of input: the distinction between continuous and discrete input. Hands-free public sinks, often found in airports, demonstrate a discrete/continuous mismatch. A motion detector senses when one waves their hands under the faucet. These devices can be difficult to trigger if one does not know the position of the sensor, or improperly judges the required distance. These faucets undergo a discrete state transition based on a continuous proximity sensor — such devices are difficult to trigger once, and even harder to predictably start and stop (luckily public faucets run for a while after being triggered). Discrete and continuous input are appropriate for different settings.

discrete	**continuous**
deterministic	sensitive
low bandwidth	high bandwidth
crisp	fluid
trigger state transition	ideal for spatial variables

The crux of the continuous vs. discrete issue, as will become clearer with subsequent examples, is a question of knowledge — how does a user know what action will result from a given input? The closing of the tongs is a discrete input — the sensor reads a binary value: closed or not closed. The position of the tongs is a continuous input variable, and difficulty arises in the transition zones between empty space, and in the transition zone between one object and another.



The yellow zone shows the region where the tong-close input activates a Reframe-Object action on the cube. On the border of this region it is difficult for the user to predict what will happen when the tongs close. Transition zones can also exist between two objects — note the fuzzy area between the cube's zone and the pyramid's pink proximity region. Misjudging a transition zone is particularly damaging to a user's psyche. Beyond lost time, errors are a very frustrating thing, as is the feeling of not being in control which follows. One way to reduce this frustration is to use the display to tell a user what action they will be performing next.



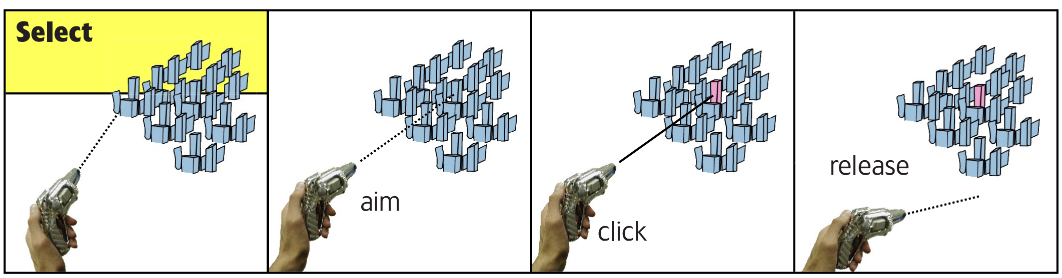
With this alteration, in the eyes of the user, the action that follows a given input does not depend on a continuous variable (tong-object distance) but rather on a discrete variable (presence or absence of the little black line). Audio cues (another form of display) can be used in a similar way to discretize a continuous interaction metaphor. In Chapter

7 we will investigate ways to further discretize this metaphor. For the meantime, we consider another input device which is mapped to a different class of actions: Select and Group.

SELECTING POINTS WITH THE RAYGUN

Using tongs (or small pincers) to select a point in immersive space is surprisingly difficult due to occlusion and tracking imprecision which yield a less coherent sense of depth than is present in natural settings. Say we are trying to select (or *pick*) one building from a 3d model of a dense city. Doing so with the tongs requires scaling the building so that the tong does not grab two objects, and also bringing the city close to the viewer. This type of preparatory reframing might be appropriate if we are slightly adjusting a building's position, but if we are merely selecting it to change its properties, or selecting it so that a Reframe-Selected action can pluck the selected object from a dense scene, there is a more direct path to our goal in the form of the raygun.

The raygun is a physical gun icon (e.g. a toy gun) with a digital beam. This tool can select a single item from a crowded city model that lies at a distance.



The raygun is more precise than the tongs or stylus because (i) the physical tool is not coincident with the point of selection, removing occlusion problems and (ii) the selection device itself is virtual, not physical. This latter aspect derives from the raygun being an input device which spans both natural and digital space. One of the claims in this text is that accurate tracking is not necessary for effective immersive interaction, and one of the design choices that removes the dependency on accurate tracking is a focus on virtual-virtual interactions

virtual-virtual relationships	physical-virtual relationships
accuracy depends on incremental tracking	incremental and absolute tracking invovled
mutable visual representation of tool	closer connection to physical body
occlusion does not affect alignment	occlusion can make alignment difficult
offers more control	appears more direct

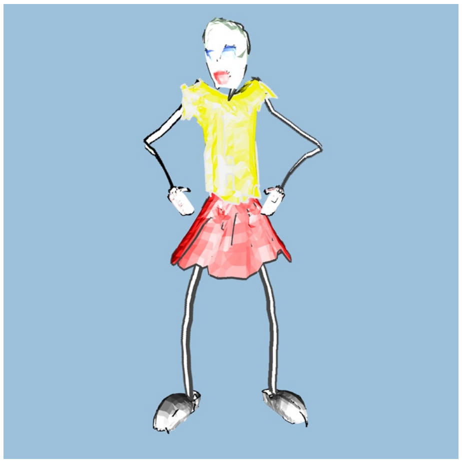
Virtual-virtual relationships depend only on incremental tracking, which greatly reduces tracker calibration requirements. In the case of the raygun, the tracking accuracy between gun and ray is not as important as the deterministic (and moreover continuously monitored by visual feedback within a coherent digital space that is not subject to occlusion difficulties) ray-scene relationship. The digital side of this relationship adds robustness, while the form of the physical device adds intimacy and control via its affordance. The raygun benefits from a clear kinesthetic link to arm direction and pointing, as well as a cultural legacy of aiming guns at targets.

Small motions of the raygun result in large motions of its beam, magnifying orientation-tracking errors. When the majority of action happens within the user's arm reach, this is a minor issue. As shown in Chapter 3, the majority of spatial-construction interaction happens within arm's reach — users can extend their arm almost directly to the point of beam-object contact to quickly reduce this distortion.

USER STUDY: ARRANGING OBJECTS IN SPACE

This section presents the first of three user studies comparing 3d interfaces to their 2d counterparts. At its heart, this text asks whether more effective, fluid, and intuitive interfaces can be constructed for spatial construction using immersive displays and 3d input devices. This study addresses the question by making comparisons with the status quo. The point is not to claim that the interfaces presented here are the best possible setups (indeed, there is a significant variation in interaction particulars throughout this text, which moreover presents partial solutions on the path to a complete paradigm of 3d interaction). The aim is to show that there are certain emotional, conceptual advantages to this style of interaction, and this point is demonstrated primarily by looking at user's written reactions to the interactions.

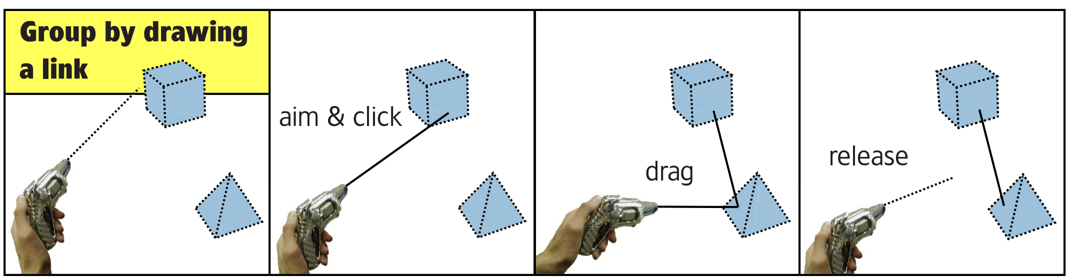
The study below uses the two devices presented in this chapter, the tongs and the raygun, and in addition a third device (a lightsaber built upon the handle) which is described in depth in the next chapter. The task is to take a set of body parts in a row and arrange them to make a digital character.



An assembled character

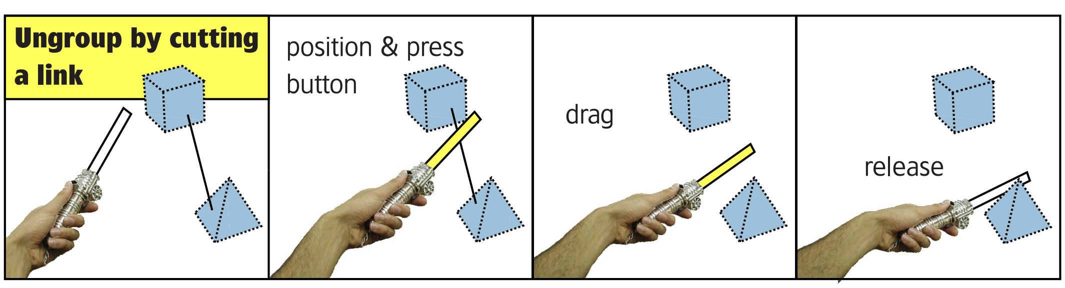
The task in the user study is to take eleven body parts (two hands, two feet, a skirt, a torso, a head, two arms and two legs) and assemble them to form character, as shown at left.

The tongs activate the Reframe-Scene and Reframe-Object actions as described above (Reframe-Component is used in the DNA modeling task described in Chapter 7). The Raygun is used to Group objects by drawing a link between them



When objects are linked together, their groups are merged and a line is drawn between the two objects to represent the group. The clicking pattern shown in the above cartoon came after some experimentation. Another model is to click on the first object, release, then click on the second object. This phrasing makes it easier to cancel the operation — if the user releases the button after clicking on the cube, the operation stops, and the user can make another selection.

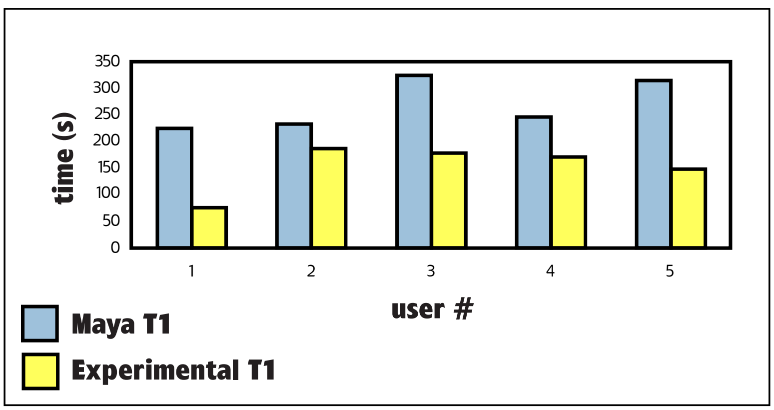
The lightsaber (a combination of the physical handle and a digital beam) is used to cut these lines and Ungroup objects:

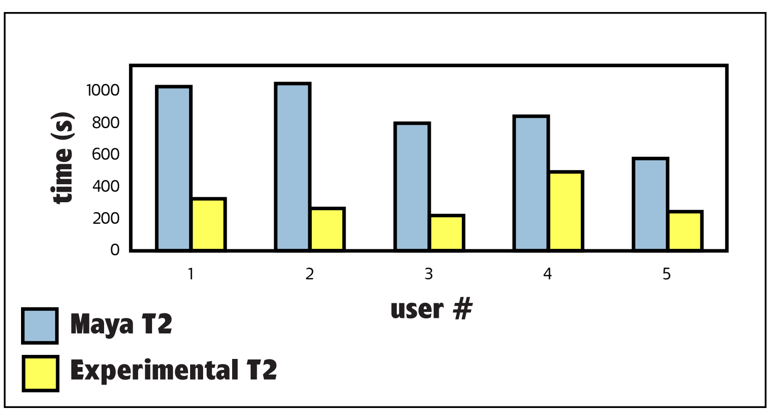


Five sample users (one of which had experience with modeling software, none of which had experience with virtual reality), used both the experimental 3d interface and Maya's interface to complete two tasks. Maya uses the arcball rotation controls and the translation widget illustrated in Chapter 3. When a modifier key is held on the keyboard, one mouse button rotates the camera, and another one pans the scene. The first task (T1) is to combine the pieces to form a skeleton. The users were specifically asked to pose the arms so that they were not parallel to the torso. This forced the users to rotate the hands. It was in this rotation that the Maya users encountered the most difficulty.

Task 2 (T2) consists of a rearrangement of the shape built in Task 1. The users were asked to move the shoes so that they were bound to the wrists, and the hands to the ankles. This required the Ungroup action which breaks bonds. Since the users posed the arms gesturally, this command forced them to rotate both hands and feet.

Task completion was faster in every case in the experimental system. The data suggest (although are too few to prove) that the experimental system allows objects to be placed much more quickly than the commercial state-of-the-art.





The affective, emotional part of a user's experience is as important as speed and productivity. Direct quotes from the user surveys qualify the experience. Users had a lot of difficulty getting a sense of space with Maya:

> *To visually determine the location of a shape I had to rotate the screen several times; it was hard to translate the 2d picture into 3d.* [quote from user #1]

> *Maya system was frustrating and difficult to really see what I was working with. In the experimental system I never had this problem. The experimental system is superior to Maya in every way.* [#2]

The following user's observations are a concise argument for the value of a strong connection between body and space that I call a **direct union** (more on this subject later):

> *The experimental system allows me to move objects in 3d space in a more intuitive way than Maya. When I want to move an object from point A to point B in the experimental system, I just need to grab it and move my arm along the direction from point A to point B. By contrast, with Maya, I have to think about the displacement along X,Y, and Z axes separately. This is less intuitive, and I often need trials and errors to finally reach the target position.* [#3]

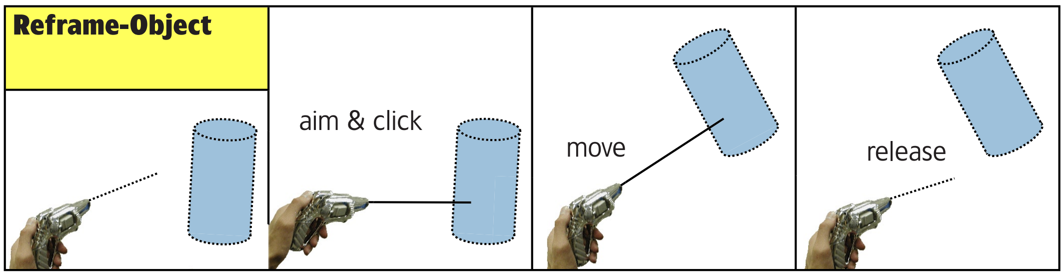
Even though the experimental system does not haptically render forms, the tangible input devices lent a sense of physicality to the system:

> *It was very intuitive how to use the experimental program because it was almost exactly like putting together tangible objects. [#1]*

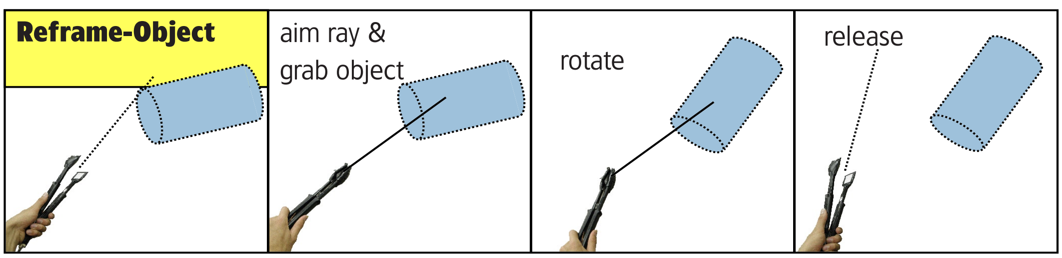
CONTRADICTIONS

There are some thoughts in this chapter that the observant reader will find to be contradictory. On the one hand, there is the notion that point-selection is bad in immersive space, and the raygun is a better way to do selection. Then there are the tongs, which grab objects in a manner that is similar to point selection. What is going on here? To understand, and resolve this contradiction, one must think about interfaces in fuzzy terms that do not leave the reader with an absolute conclusion. In interface there is never one best way to do things. Some users like to copy files via drag-and-drop with the mouse. Others copy and paste the files using the keyboard, with keystrokes to shift from the source window to the destination window. Other users prefer a command-line interface for the same task.

Note that the raygun can be used to Reframe objects with an interaction that looks somewhat like this:



This interaction eases the selection, but greatly reduces the amount of rotation that can be performed during a Reframe action. The tongs can select their targets with a ray-style metaphor



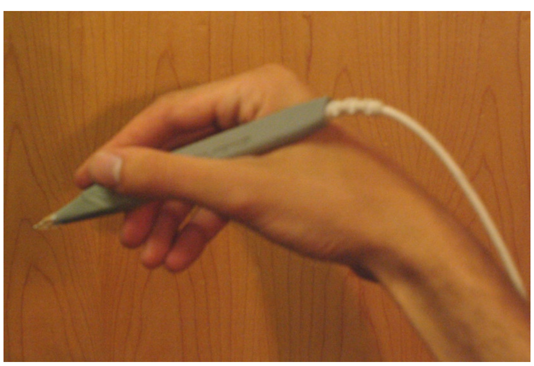
Note how, after the tong rotates and before it releases, the ray is no longer parallel to the tong. This interaction eases selection, and maintains an ease of rotation, at the cost of making the mapping between the cultural affordance of the tong and their function in the digital space less direct (although note the tong can represent the pointing direction surprisingly well). Interface design is not cut and dry, there are no unanimous preferences for interface style. Chapter 7 demonstrates a method that supports personalized mappings: users can make rayguns grab and tongs point as they wish.

6 freeform tracing

creating shapes with the hand and handle, deformations, and more user feedback

CREATION is an important aspect of constructing spatial forms. The tongs and raygun can arrange things, but can they create geometry? They are built for grabbing and pointing; their affordance does not map as well to the class of actions that involve tracing out paths in space. Dragging a tool through space while active is both particularly easy in the immersive setting, and also particularly useful for actions ranging from freeform sketching, to freeform deformation, cutting, erasing, and curve tracing.

In mouse-based interfaces users typically create curves by specifying a series of points which the desired curve interpolates. This is in line with the dominant metaphor for 2d mouse interfaces: placing and moving 2d points in space. Holosketch [Deering 1995] demonstrates that curves can be created by directly tracing their path in 3d immersive space. The physical input device in Deering's system is the 3d stylus that commonly ships with motion tracking systems such as those made by Polhemus and Ascension.



3d stylus

This pen-shaped tool is commonly used in virtual reality applications.

The physical form of the stylus is derived from the pen, a shape that is designed to be pressed against a surface as it is moved. This device suffers from two shortcomings:

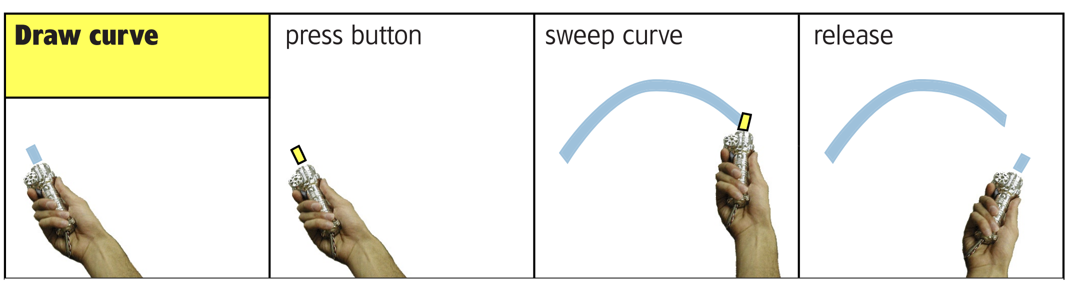
• The stylus affords a precision grip which is not appropriate for body-scale interactions where large muscles are used to control tools. Moreover, low precision tracking calls for interactions which are not dependent upon finevz manipulation. In these environments, users frequently switch to a power grip (observed in my early experiments), and a tool which supports this position is more appropriate.

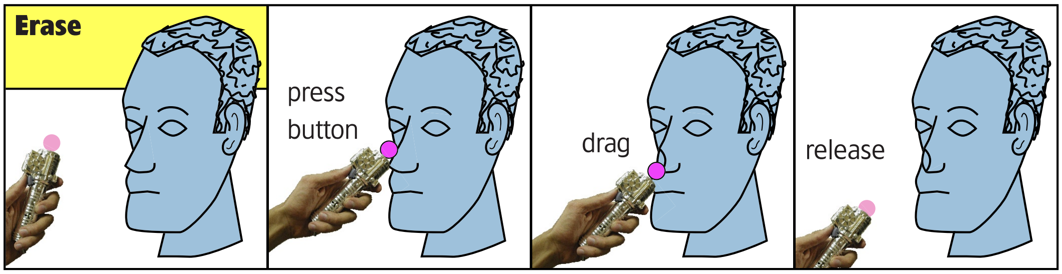
• While there is a cultural affordance linking the pen to tracing shapes in space, it is less appropriate for other tasks such as erasing and cutting. The device below, which I call a **handle**, has a more general cultural affordance.

3d handle

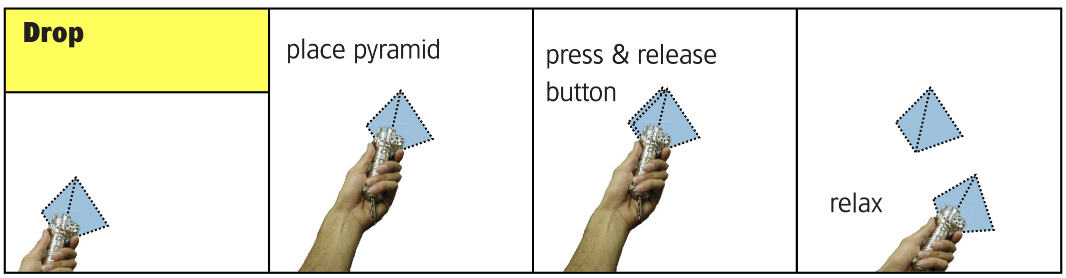
This form, built by adding sensors to the handle from a toy lightsaber, has a more appropriate kinesthetic frame and a more flexible cultural affordance.

The handle is held the way one holds a briefcase, a glass or mug, a rope in the game of tug-of-war, a tennis racket or a baseball bat (when held with one hand) — i.e. with our opposable thumb well utilized. The affordance of this tool indicates that something is being held, and as such supports a digital effector which can be carried through space to perform actions such as tracing a curve, and erasing (erasing is essentially tracing a curve in negative space).





The handle is also a good choice for placing elements in a scene. The cartoon below shows a Drop action. After placing one pyramid, the handle is ready to drop another one.

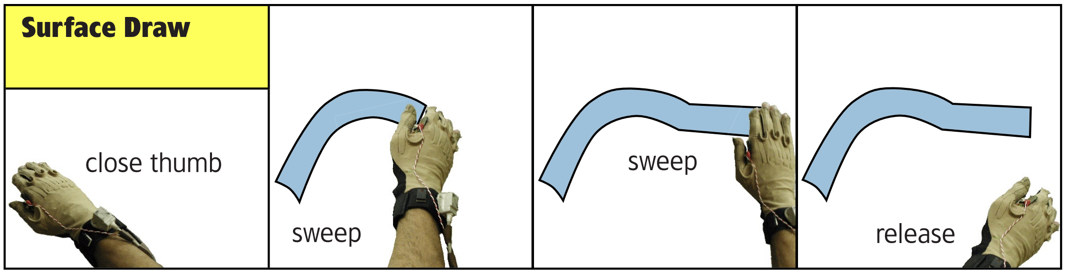


Here are frames from a user erasing a hole in a model's head. Note that this view is not from the user's perspective, hence the eraser ball's icon does not match up with the handheld tool.

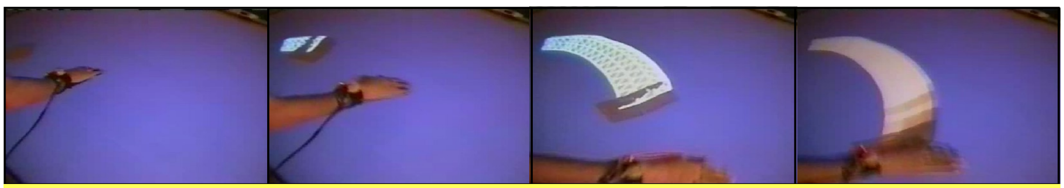
Erase: dragging the eraser tool through space makes a hole in this form.

EVEN HIGHER BANDWIDTH WITH THE HAND

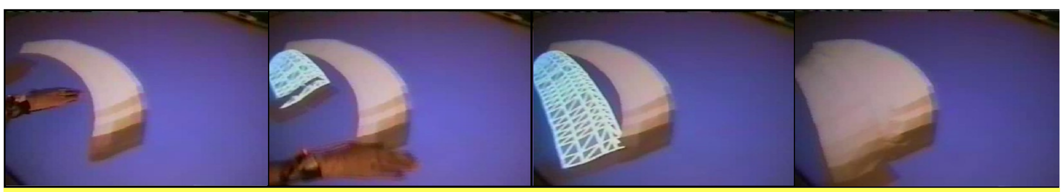
The handle enables one of the chief advantages of 3d interfaces for spatial construction: a large amount of data can be input in a short amount of time. The hand, sensed with a glove, can be used to input even more data, and this is quite useful for a richer curve-tracing interaction that specifies several points along the path in a band of surface as the curve is traced through space, forming a stroke.



The images below show an implementation of this interaction. The second row shows strokes merging together to form a larger continuous surface, as in [Schkolne 2001].



Surface Draw: the hand paints a stroke in space.



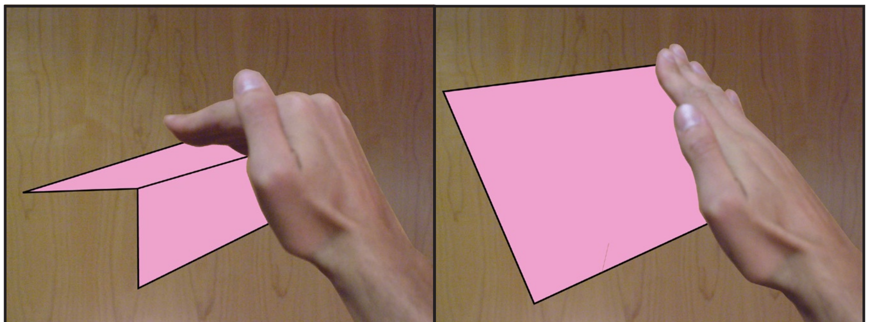
Two strokes automatically merge to make a continuous surface.

The video captures do not show the geometry of the scene accurately, since they are filmed from over the user's shoulder. This composite image shows more clearly the link between hand and form that the user experiences:

Surface drawing

The path of the hand in space is rendered as a geometric object. The curvature of the hand defines the curvature of the stroke: a large amount of data is specified in each sweep of the hand.

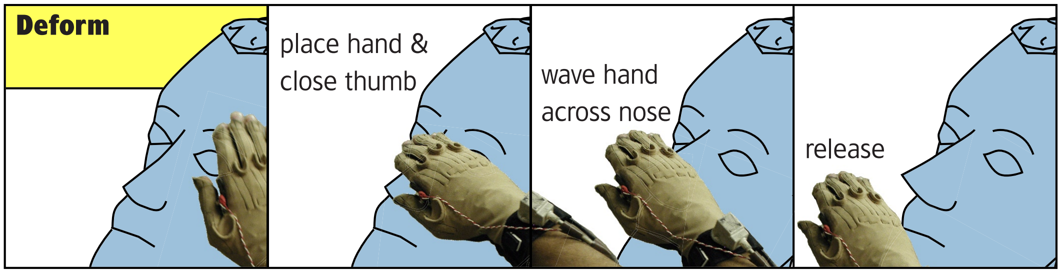
In addition to freeform curved strokes, the hand can also be mapped to constrained strokes that are perfectly flat, or have exact right angles in their geometry. This is done by snapping the samples on the hands to exact flat surfaces if the hand is approximately flat. Here a right-angled curve is drawn by holding the hand at an approximate right angle:



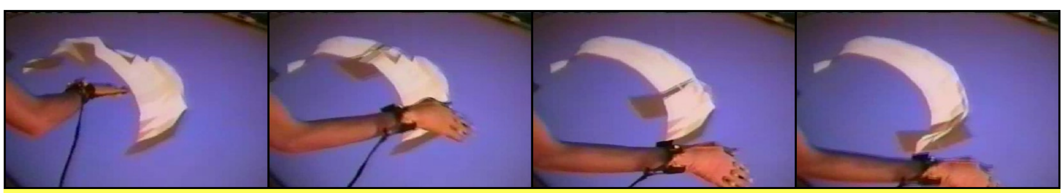
The curvature of the hand controls the curvature of the stroke – here the computer snaps approximately right-angled and approximately flat hand curvatures to perfectly right-angled and flat strokes.

The hand is particularly good at specifying tangent planes. While the handle has a specific orientation in space, its kinesthetic framing does not map naturally to this orientation (by this I mean that the proprioception of this orientation is not direct, there is a weak relationship between body position and orientation of the handle). The hand becomes a plane when flattened out, and thus specifies a tangent plane in a way that is directly perceived by users, establishing a natural kinesthetic frame around itself.

The tangent plane can be used as a variable to control a Smooth action (where the surface moves closer to a flat surface whose normal is dictated by the hand's normal) and a Deform action (where the surface moves in the direction of the hand's normal).



Many methods can be used for smoothing and deforming surfaces [Sederberg 1986, Taubin 1995, Desbrun 1999]; in practice the method deployed depends on the surface representation used. Much of the research on smoothing meshes presents highly specialized, quite sophisticated methods geared towards the non-interactive session. In practice, we can implement a simple method that allows both smoothing and deformation to take place. The images below show the results of an algorithm which moves each vertex of polygonal mesh slightly toward the hand when the tool is active, with less deformation applied to points far from the hand. The direction used is the finger's normal (the normal of the curve that was used to draw strokes in the previous images). Sweeping the hand through the middle of the stroke smooths it:



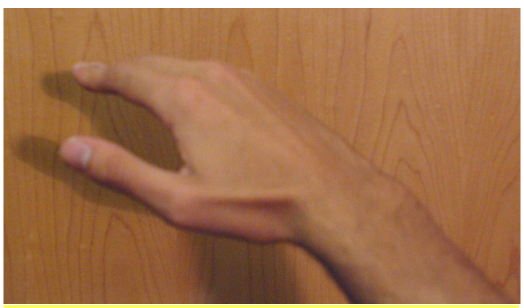
Smooth: A surface is polished by rubbing the hand over it.

Sweeping the hand through the surface has an averaging effect – the vertices move to the average hand position over the duration of the sweep. This algorithm can also deform a surface. If the hand remains slightly above the surface, introduces a slight bulge, as seen below:

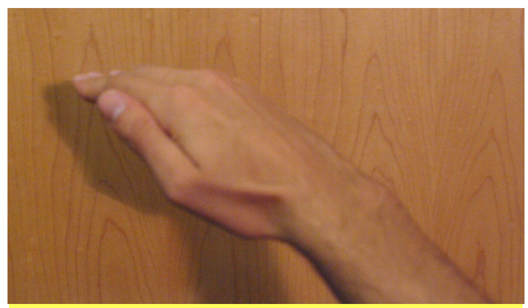
Deform: The surface is slightly altered by rubbing the hand over it.

Similar stroke-based deformations are available through Maya's Artisan package. This is a 2d interface that maps strokes in screen space to model deformations, applying consistent rules to determine the normal vector of the deformation Gaussian (e.g., the surface always moves in the normal direction). Note that there are advantages to this system, in that there is a consistency to the direction of deformation. Such a consistency can be mapped into a hand-based tool, but the immediacy of the link between hand-space and deformation-space disappears. It is my intuition that the extra control offered by the hand (where the direction of deformation mimics hand orientation) provides a wider range of deformations.

There is no established method to trigger the hand's action. There is no inherent button on the hand. I have found the angle of the thumb to be a successful trigger for the thumb's action. An **open** thumb is the relaxed state; when the thumb is pressed against the base of the thumb I say it is **closed**.



This thumb is **open**.



This thumb is **closed**.

In the early versions of this system, the continuous joint angle sensors of the Cyberglove (an 18-sensor instrumented glove) were used to detect this difference. Users had a lot of difficulty with this mode of sensing, both accidentally triggering hand actions and accidentally stopping them mid-stroke. This sensing was refined twice, first with a contact sensor which clarified the motion required to trigger input, then with an actual button which further eased the input. The button also provides a haptic display — the user can sense when the button is clicked.

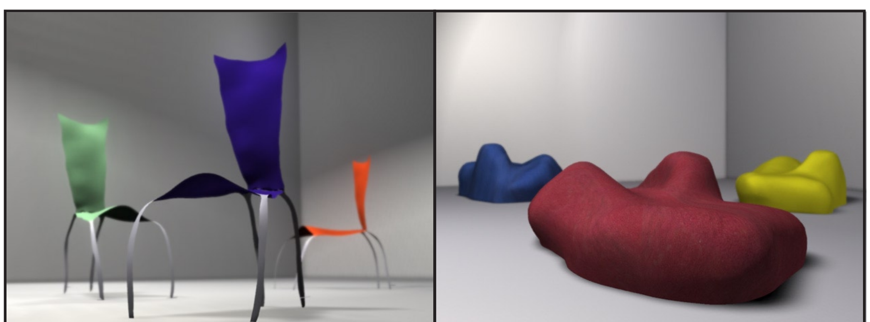
Discreting thumb input

The Cyberglove was augmented first with a contact sensor (the black patch at base of index finger), and later a button (white form on top of patch). Both modifications improved the sensing of the thumb close input.

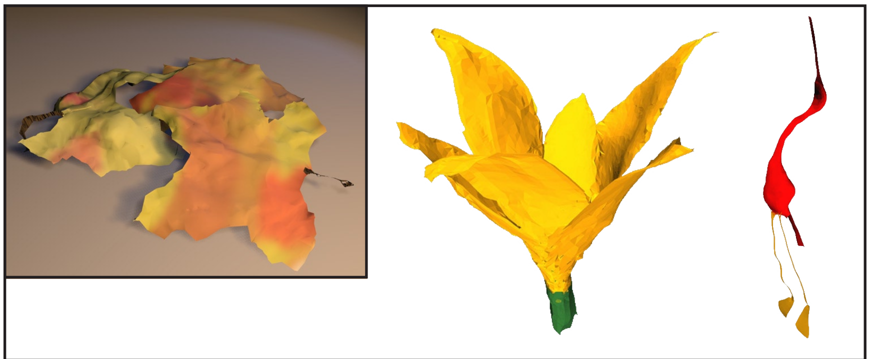
These issues further demonstrate the discrete vs. continuous issue: mapping continuous joint angle to discrete mode transition confuses users, while mapping a discrete button press to a mode transition is simple. With the addition of a button, mode is crisply displayed to the user via tactile feedback.

DRAWING SURFACES WITH THE HAND: RESULTS IN PRACTICE

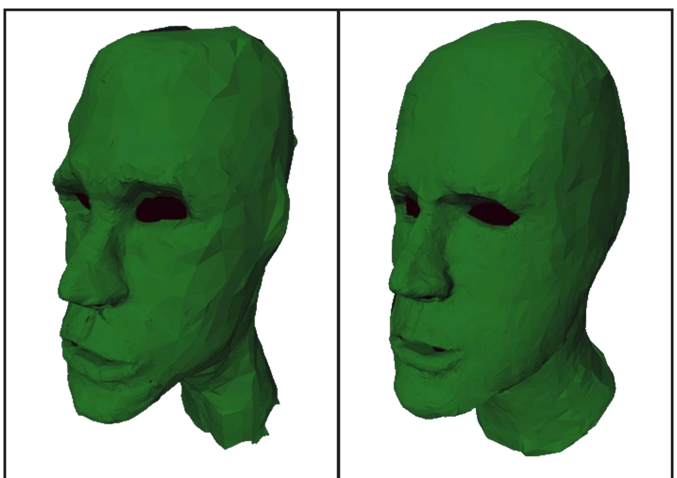
In 1998 and 1999, along with artist davidkremers, I experimented quite a bit with this medium. The results shown belown were created by myself unless otherwise noted.



Each of these furniture designs was created in about an hour. The sofas at right have been smoothed by a mesh processing algorithm. The smooth organic shapes seen here are well suited to this method of creation.



The natural shapes shown above exhibit a roughness that is difficult to achieve with ultrasmooth spline and multiresolution representations. The shakiness of the hand becomes the roughness of a surface. Although this process is non-haptic, the element of touch has significance. While not volumetric, and more rooted in construction than editing, surface drawing has many of the tactile elements of clay sculpture. The leaves at left were created by davidkremers.



These two versions of a head model show the effects of the smoothing operation. Shown at left is an early model of the head. At right, the smoothing/deforming feature has been used to correct the proportions of the face and smooth the head's surface.

Another advantage of unstructured metaphors that relate so clearly to physical space is that they are somewhat open to interpretation. Artist Jen Grey worked with the system, in collaboration with Sheri Bernham, over a period of several months. At times they chose to work by tracing human models that posed in front of the display, paying little attention to the output until after the process was complete. Their unique approach yielded a very different geometric vocabulary, as seen in the images below.

The image at left shows a series of traced shells of human bodies. Done without looking, these have an exagerrated roughness to them. The Centaur shown at right is a more deliberate form. The highlights on the surface of the form were created later with rendering software. Both images created by Jen Grey.

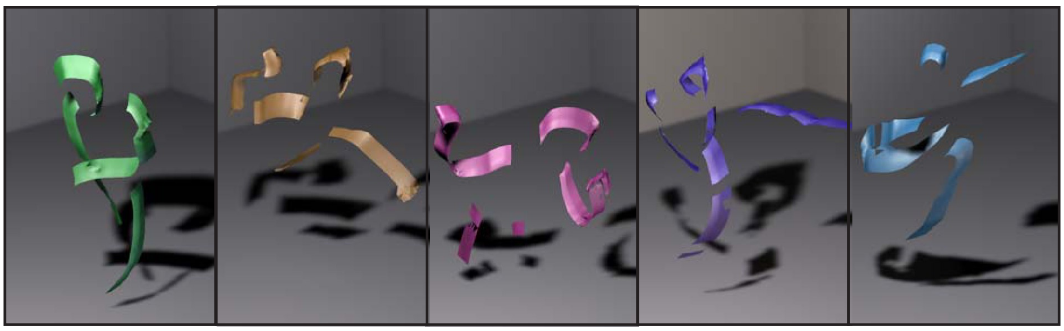
Grey and Bernham were concerned with gestural qualities, and even took interest in the noise inherent to the trackers. Overall, they saw this as an opportunity to expanded their artistic vocabulary and experiment with new process.

> *To see abstract images pour like water from my fingertips is sensational... Even more amazing is to see what touch looks like!* [Grey]

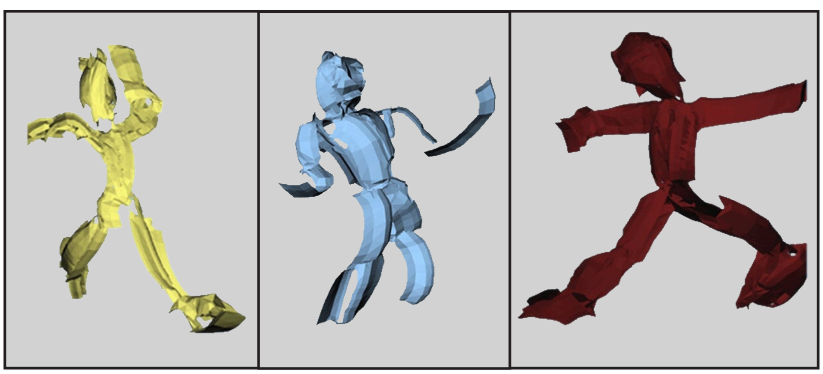
> *Much like when paint programs liberated drawing in 2D on a computer, this system liberates the normally rigid/structured process of building computer models in 3D space. Gesture becomes important again.* [Bernham]

RAPID CONSTRUCTION: 3D GESTURE DRAWING

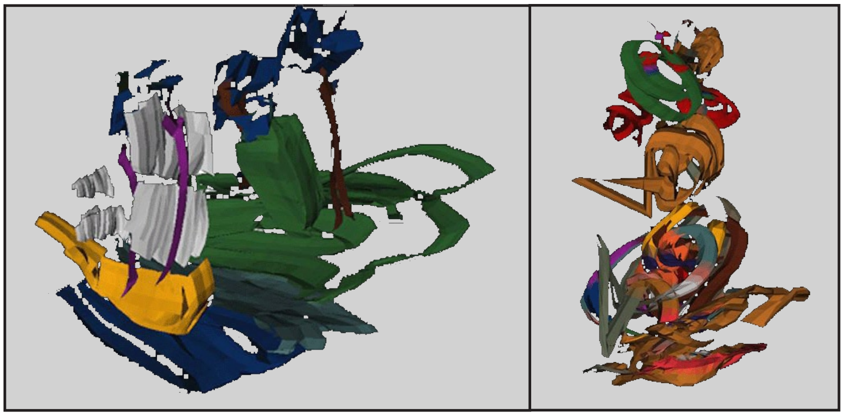
Traditional 2d artists often sketch very quick impressions of posed human models. These low-detail, highly emotive drawings are known as gesture drawings. I experimented with 3d gesture drawing to assess the fluidity of the interface, and its suitability for quick conceptual drafting. An early surface drawing interface, in which the stylus was used to move the world (before the tongs were invented), I made the following sketches:



Each of these drawings was made to match a model that posed in front of me as I drew. About a year later, using a newer interface which included the tongs, i repeated this experiment. Here are some results from that session:



This ability to create form quite quickly was seen, not only in my own experiments, but also in those of users new to 3d interfaces. While most visitors who visited the lab made forms of no consequence, two large exhibits had audiences large enough to include a few talented artists who made interesting shapes almost immediately with this interface. The following forms are from these exhibits, the image on the left coming from the SIGGRAPH 1999 Emerging Technologies exhibit, and the image on the right originating in the 2000 Mostra da Realidade Virtual in Rio de Janeiro.



Neither user spent a long time with the interface — each shape above was made in about ten minutes of total use (including learning time).

FREEFORM SURFACE DRAWING, A USER STUDY

The quality of the two creations above warranted further investigation. Eight art students with experience in pencil drawing, 3d modeling, or both were recruited for a study to investigate both their ability to create forms immediately, and also to find out how they related artistically to these interactions. They were presented with a small toolset consisting of three interactions:

Reframe-Scene	using the tongs (one tong moves & rotates, two tongs together scale the entire scene)
Erase	using the handle (yellow silicone form, illustrated in Chapter 9)
Surface Draw	using the hand, sensed with a CyberGlove

Strokes were merged together as shown in the previous section.

After being briefly introduced to these interactions, they were asked to draw a practice doodle (5 minutes), a flower (5 minutes), and a human (20 minutes). Our observations should be tempered by the consideration that these users had no prior experience with constructing objects in immersive digital space. They were given the task of learning the interface and constructing some very demanding shapes in a short amount of time. Indeed, most subjects were intimidated by the task of drawing a human figure in this brand-new creative space. Like an artist picking up a pencil for the first time, these novice users could not make many of the mature observations that require experience. This user study tests the freshness and intuitiveness of surface drawing at first glance.

Here are some of the shapes they created within their first half hour of exposure:

Two results from the user study, drawn during the five-minute flower drawing task. The flowers on the left live in 3d space, while the shape at right is much flatter. The users had widely varying degrees of spatial complexity in their creations.

Two bodies drawn in the twenty-minute figure drawing task.

The subjects were given a survey in which they rated the usability of the interface. The tongs were found to be very usable (average rating 4.6 on a scale of 0–5). The display was rated fairly usable (average rating 3.9). The glove and eraser were found to be moderately usable (average ratings 2.8 and 2.6, respectively). Artists with a background in traditional media seemed to appreciate the system the most. One artist described his experience with the system as:

> *Fun. Takes a while to get some tactile fluency but one senses with plenty of practice one could get to be quite capable. The tools supplied are really quite versatile, and one appreciates that with greater familiarity one would be able to make some good art.*

An artist who was most familiar with pencil drawing and had additional experience with Maya enjoyed the 3d interface:

> *I was completely amazed at how quickly I interpreted and understood the canvas and model to be existing in space. It was immediate.*

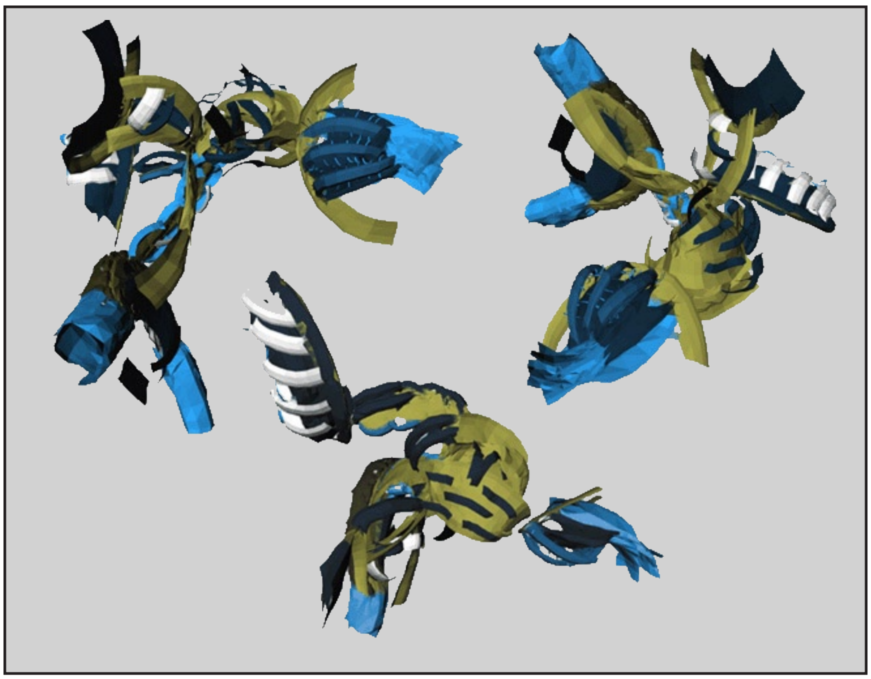
Users who were most familiar with spline-based 3D modeling software sought their familiar control handles, as exemplified in the response of one seasoned Maya user, stating that the 3d interface "needs finer control (or long-term training) for anything to be done seriously." Everyone seemed to like the tongs: "Tongs are super cool for providing quick access to all parts of the figure drawn." "Surface Drawing is unique in speed of creation and control over figures once drawn. Other media are slow and awkward to manipulate in comparison."

We were surprised by how few artists worked three-dimensionally. For example, many of the figures were not inactive poses, but rather people standing as if they were lying against a wall. The flowers, which were perceived by the subjects as a less daunting task, exhibited more three-dimensionality and playfulness. All users had difficulty with the glove, which in this early study still used continuous joint sensors to determine if the thumb was closed.

The user study shows that artists are comfortable in 3d immersive space and appreciate its benefits. Many users acknowledged that more practice is necessary to become accomplished with the system, although they did not have problems learning the basic interface. Many of the users wanted to spend more time using the medium. There were enthusiastic remarks: some users asked when they could have one in their own homes. More info on this study, and the artistic merits of interface based on surface drawing, can be found in related publications [Schkolne 2001, 2002]

TOWARDS SPATIAL COMPLEXITY

One of the hypotheses of this investigation is that working directly in 3d space, with a material method of construction unlocks a deep spatial intuition, a form of thought in the sense of Arnheim [Arnheim 1969] that allows humans to understand space in a deeper sense. My experiments in the construction of abstract shape showed a certain type of spatial sophistication that is exhibited by the following image:



This picture shows, not three objects, but rather one object from three different perspectives. Seen on the flat page, it is difficult to imagine how the 3d form sits in space. Yet in immersive 3d, as the shape is made, this intuition is clear and the spatial placement is obvious. It is this distinct advantage of 3d interfaces that led to the experimental application that will be discussed towards the end of the next chapter: DNA construction.

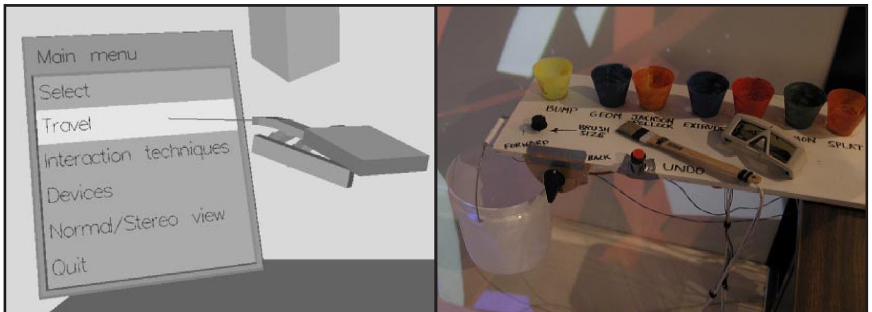
7 accessorizing phicons

controlling a multitude of digital tools with a few phicons; an application in DNA modeling

THE SPECIFICITY of the tongs is high: they are tailored to a single task the way non-computational tools are. A mechanic works with a series of wrenches, welding tools, jacks, screwdrivers, etc., each is customized to a certain task and displays a highly specialized affordance. There is a richness, both aesthetically and psychologically, to this identification of form and function. My early vision was of a world with a whole host of wireless input devices that would sit in the equivalent of the mechanic's toolchest. A few tools would be selected to do each task. This was partially inspired by mediaBlocks [Ullmer 1998], which associate a physical form with data. In this approach the identity of each physical tool is closely linked with the function it performs.

Task-specificity is decisively at odds with the culture of computer use: users expect to quickly switch from task to task. When I use Photoshop I frequently use obtuse keystrokes to change from the Marquee, to the Eraser, to the Brush, to the Zoom tool and Hand tool for navigation, and so forth. While the often arbitrary nature of these keystrokes offends an inner sensibility, I cannot argue that they greatly speed operations within software and lend an amount of fluidity to the interactions of the experienced user.

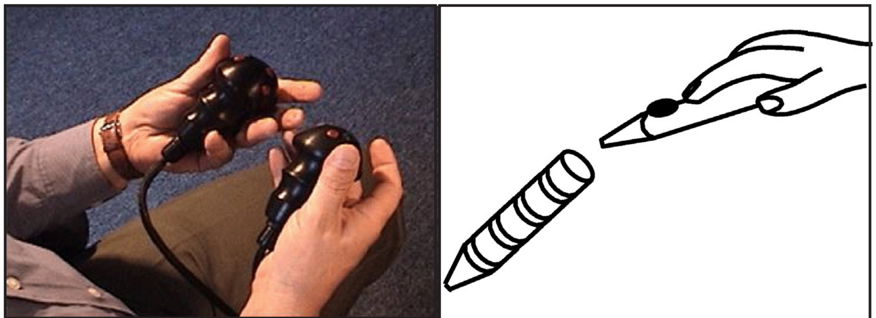
The virtual reality of the 1990's quite often used a stylus to perform multiple functions. For example, Serra et. al [Serra 1995] use a toolbar to choose the stylus' function from an array of tools including a generic pointer, a line cutter, and a control point mover. Their toolbar has icons that can be selected by placing the stylus over an icon, or rolling a linear slider along a toolbar. In their system, changing the tool changes the complete display of the tool (the stylus and hand cannot be seen in their mirrored display setup). The floating menu, an adaptation of the traditional 2d menu, is found in the work of Angus and Sowiter [Angus 1995, see also Kim 2000]. In some implementations of this method, the menus are selected by dragging a 3d cursor over the menu. Other approaches use ray-casting to select from these menus. In CavePainting [Keefe 2001] tool configuration is changed by dipping the brush into small buckets. This method does not require precise selection, and allows users to remember menu item placement with physical memory.



Using ray-casting to select from a floating menu [Flasar 2000], and selecting from a physical menu of buckets in CavePainting [Keefe 2001]

Another approach to remapping 3d tools is to place buttons or other features on the device itself. This method reduces travel time and ideally makes it quicker to select items. The meaning of each button can be very difficult to remember, as the mapping between buttons and functions can be quite arbitrary. These multibutton approaches are at odds with a line of thinking epitomized by the Apple Macintosh interface designers' decision to only support one mouse button (instead of the two or three that are seen elsewhere) [Apple 1992]. The Macintosh philosophy is intended to simplify interaction, provide aesthetic integrity, and encourage more diverse inputs for actions that would otherwise be mapped to the second or third button. Whether or not this rather orthodox philosophy is indeed ideal, the issue it highlights is at the heart of the difficulty with multibutton input devices. An example of this approach is seen in the input controllers of the InDex modeling system, each of which has five buttons used for different functions. These tools, shown below, came from an earlier incarnation of this tool (the SmartScene modeler) which used pinch gloves as input devices.

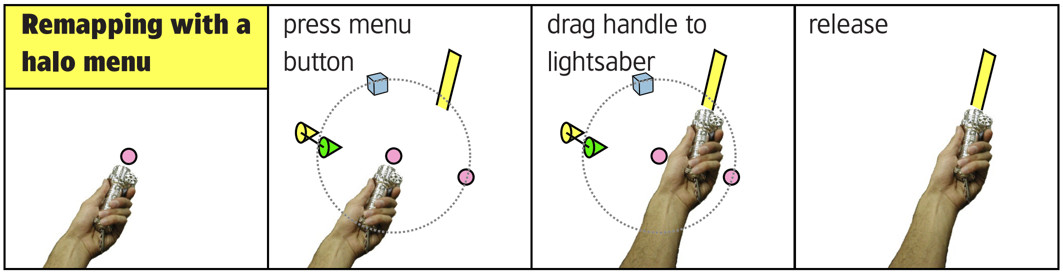
The pinch-glove based SmartScene interface raises some issues with arbitrariness of inputs that are also seen in my experiments with using hand postures to activate different surface-manipulation modes (described in Chapter 9). A quite unique approach is the ToolFinger, which actually presents multiple tools simultaneously. As seen below, the ToolFinger is a visual element that appears a small distance from a sensed stylus. Each band of the digital finger is a different tool — intersecting the first band with a surface



The multibutton tracked controllers used in the Digital ArtForms' InDex system (left), and the ToolFinger interface [Wesche 2003].

control point and pressing the button moves it. Similarly the second band is used to delete control points. This metaphor allows all tools to be accessed at once, although it is only suitable for interactions based on a point-selection metaphor which both relies on highly stable tracking and moreover does not allow for the richer kinesthetic relationships which tools such as the hand, raygun, and tongs support.

A solution that allows a tool's digital effector to be remapped with little travel time, presents options to users in a device-specific way, and uses display to aid memory is the adaptation of the 2d metaphor of the pie, or marking, menu to 3d space. Pie menus [Callahan 1988] display a list of options around a selection cursor, speeding selection from a menu. These menus can be extended hierarchically to display a large number of options [Kurtenbach 1993]. Michael Deering used a pie menu to change the function of his stylus in the HoloSketch system [Deering 1995]. This general approach has several advantages in 2d that appear to carry over to 3d: consistently with Fitts' Law, they are quicker than linear menus for selection [Kurtenbach 1994]. These local menus do not require a user to travel towards the menu, and furthermore they can represent only the options available for a given phicon. Note that, in contrast to Deering's approach, the 3d menus described here do not fade away the scene and replace it with the menu, nor does the system wait half a second to display the menu. These menus immediately display a ring (or *halo*) of tools around the input device. Dragging the physical tool to coincide spatially with one of the options matches the physical tool with a new digital function. One of the key benefits of these halo menus is that they can be used without visual attention. Experienced users can remap a tool in a fraction of a second, while novice users can take their time and look at the displayed choices.

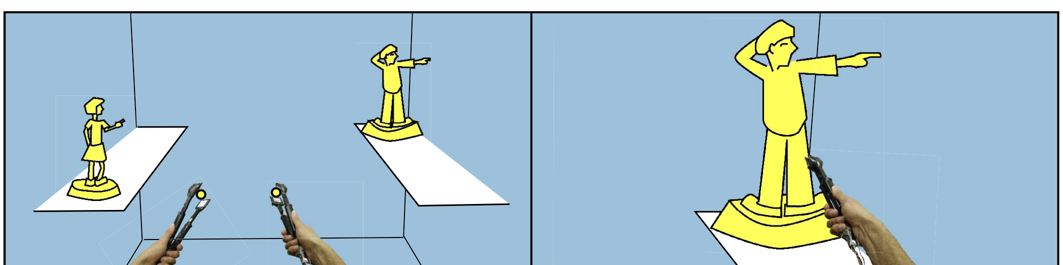


Remapping with a halo menu

One of the most useful side-effects of this menu system is that it enforces a certain concept that I have found to be most fruitful in the design and implementation of 3d interfaces: 3d input devices can be decomposed into a physical base and a digital accessory. This makes it easy to have an accessory controlled by different devices (such as the tong-select and raygun-reframe interactions suggested at the end of Chapter 5). This split also allows simple metaphors, such as the halo menu, to extend the digital toolset.

ENRICHING THE RELATIONSHIP BETWEEN BASES AND ACCESSORIES

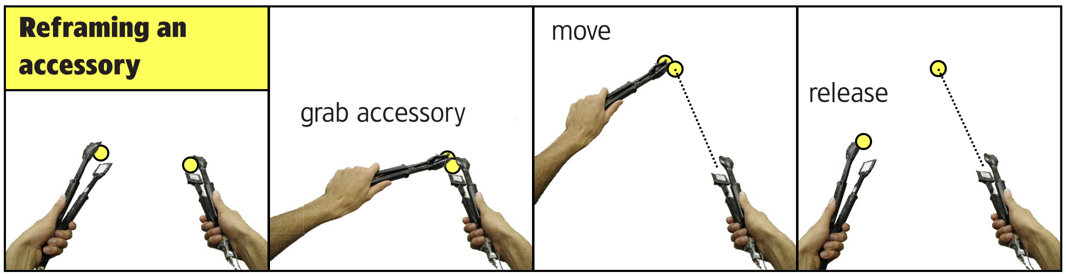
Consider a hypothetical problem: a user wants to place a statue on a shelf, and the orientation of the statue is important. Let's say a male statue should be pointing at a female statue on the other side of the room. If the statue can rotate only around the tong phicon itself, the user has to Reframe the scene to focus on the male, and the female statue can no longer be seen. While the male statue can now be easily manipulated, it cannot be appropriately aimed.



The two-statue problem: *place the male statue so that it is pointing at the female statue*. in order to move the male statue, the user has to scale the scene so that the female is no longer in view.

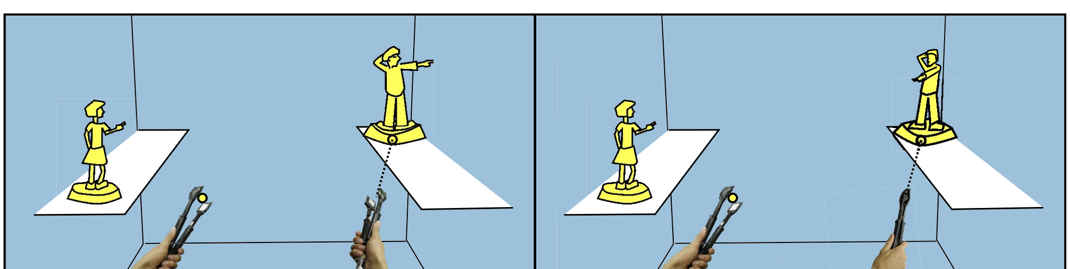
While this example is quite particular, and as a rule of thumb interaction is best within armspace, some users quite frequently want their actions to occur at a distance. In particular, one of my artist collaborators wanted to see the whole scene as he manipulated fine details.

A resolution of this difficulty comes from applying Reframe actions to the phicons themselves. For example, a tong's center of rotation can be represented as a sphere, which tong B moves some distance away from tong A. From this point on, the incremental motions of tong A are applied to tong A's accessory, and the distance relationship is maintained.



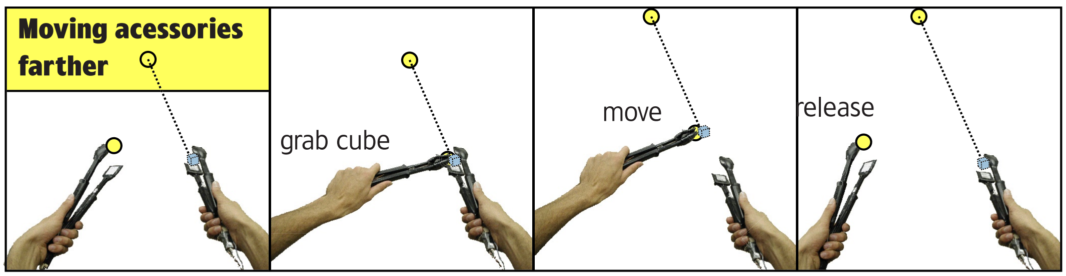
Note that, within our conceptual split between base and accessory, this change in accessory placement can be quite easily implemented. Approaches such as Voodoo Dolls, Worlds in Miniature, and Pierce's image plane interaction techniques [Pierce 1999, Stoakley 1995, Pierce 1997] do not map well to the tangible tools used here. The go-go interaction technique [Poupyrev 1997], which extends the last third of an arm's range nonlinearly, is a method that is consistent with the base/accessory metaphor.

In practice this solution was found quite workable, even though it spatially separates a user's body from its effects. Does this not go against the whole theory of tangibility by separating touch from data? I understand this as illustrating that tangibility, in these 3d interfaces, is mainly good in providing affordances which guide actions. Relative motion is most important for placement (indeed a small offset between accessory and phicon prevents occlusion and enhances selection). After this remapping, the value of cultural affordance and the sense of grabbing inherent to the tongs remains, while the physical immediacy is reduced.

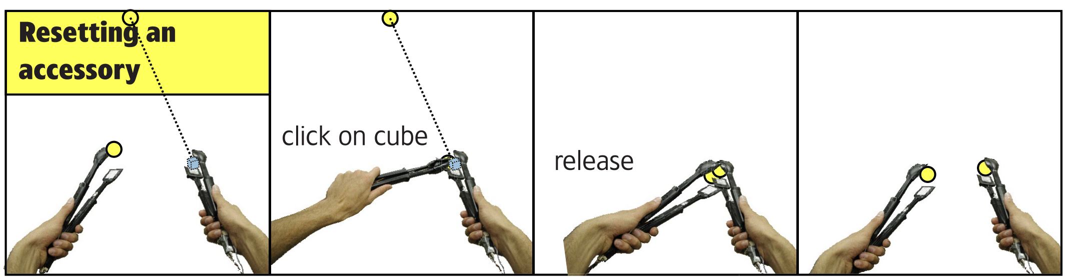


After the tong's accessory is reframed, the navigation step is not necessary: the male statue is reframed with the scene in its original perspective and scale.

In practice, users often move accessories far away from their bases. Should the user want to move the accessory even farther away, it is not possible to move the accessory beyond arm's reach. Furthermore, if the user wants to bring the accessory back to its home position, close to its parent phicon, they may have to stretch quite a distance. A cube which always lives near the phicon can be used as a proxy to move the accessory further.



Clicking once on this cube resets the accessory to its home position.



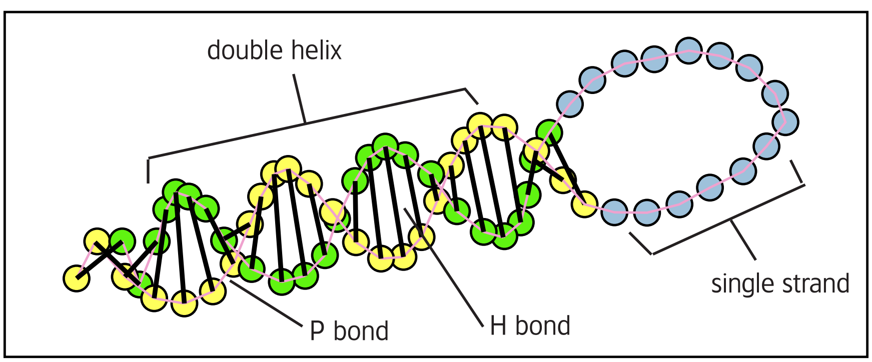
These interactions which reframe accessories are less robust than the other tangible 3d interactions we have seen. This is because of the intense point-selection involved: two tongs and two virtual objects need to come together in a precise way. The raygun could perhaps more easily perform this selection to avoid occlusion issues. This interaction is mentioned, both to suggest the utility of reframing accessories, and also to show that there is some power in conceiving of accessories as ordinary objects in digital space. Other solutions, such as placing dedicated joysticks on phicons to reframe their accessories, should also be experimented with before we decide that the methods shown above are ideal.

AN INTERFACE FOR DNA CONSTRUCTION

The local menus were tested with an interface for DNA construction. This system matches tangible UI to a structured task that does not benefit from the gestural, emotive connections that the freeform drawing system in Chapter 6 leveraged. In addition, this DNA application has more actions than the freeform drawing application, providing an opportunity to test the effectiveness of halo menus in supporting a slew of accessories on the foundation of a few phicons.

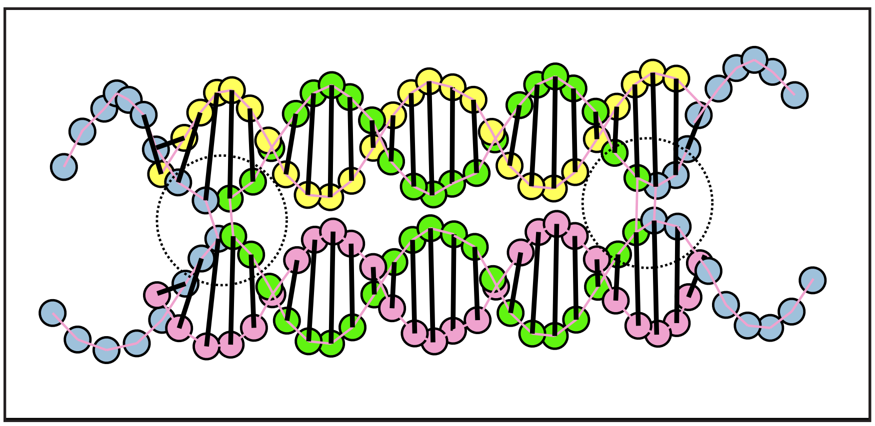
This interface, developed in collaboration with Erik Winfree's lab at Caltech, addresses a difficult spatial design task encountered by scientists researching molecular biology. The objects of study are so small that they can't be touched. They are so intricate that diagramming them with 2d paper and pencil is insufficient. 2d interfaces with stereo goggles are frequently used to view molecules, and while much progress has been made on methods to view data in virtual reality [Chen 1999], little if any work has been done on constructing scientific models in immersive space. The goal is to be able to prototype structure such as the DNA cube [Chen 1991] and Winfree's DNA tiles [Winfree 1998] in digital space to identify potential problems before they are synthesized in the lab, or to try to identify the causes of errors in a product after it has been synthesized.

This prototype interface facilitates the construction of DNA using a simplified model of bases and bonds. The following molecule shows the essential features of the spatial construction that we now consider:



DNA is comprised of bases: complementary groups of atoms that represent information in a manner similar to binary code. Although there is a standard color coding for each base, the above image uses a convention where topologically connected regions share the same color. This makes it easier to see structural features in the molecule. The yellow and green bases on the left of the image form the classic double helix. On the right is a single-stranded region that connects to the end of the helix, forming a structure called a hairpin. The pink bonds along strands are Phosphate bonds, whose structure is sometimes referred to as the backbone of the molecule. The thicker bonds across the helical axis are Hydrogen bonds.

This model is a gross simplification of the atomic composition of DNA. Each base consists of several atoms, and their bonding dynamics are in nature more complex than a discrete topological connection. There is a rich dynamic of interaction in the center of the helix itself. While in nature the bonds are continuous and can slide from one formation to another, and moreover the very structure of DNA can form in other ways, this simplification represents the majority of designs that will be considered. Another structure, of particular interest to the Winfree lab, is the double crossover molecule.



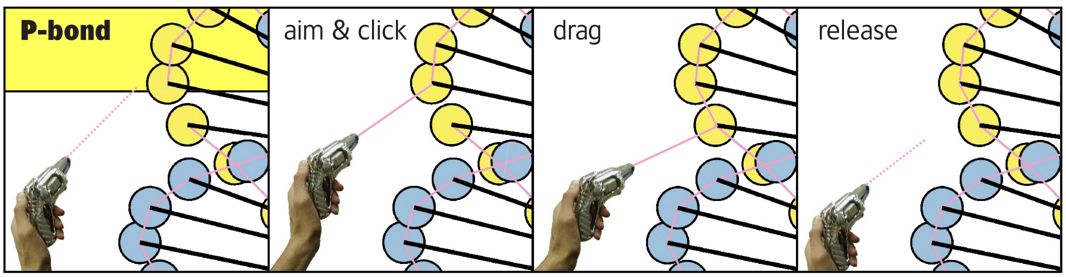
Note the two crossover regions (circled) where the helices are bonded to one another. Also of importance in the molecule are the four sticky ends. It is in these regions where one double-crossover molecule bonds with another crossover molecule. It is with these molecules that DNA can tile space [Winfree 1998b].

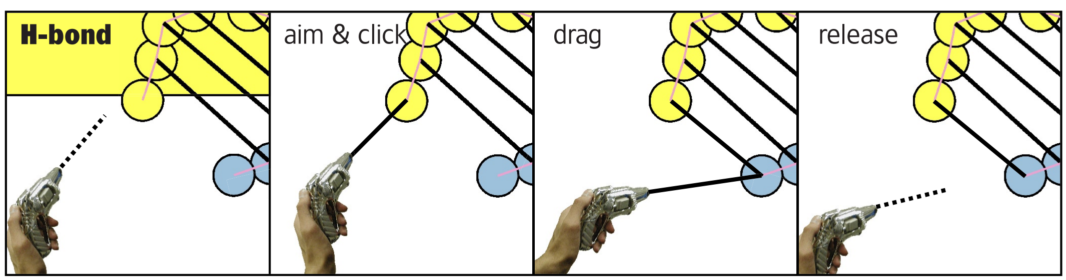
The actions for DNA construction are as follows:

Reframe-Scene	move + scale the whole scene
Reframe-Object	move a molecule
Reframe-Component	move a DNA base within a molecule
H-bond	draw a Hydrogen bond across the DNA's helix

P-bond	draw a Phosphate bond along the DNA's backbone
Cut	remove a bond
Draw Helix	creates a double-helical region at a certain position in space
Draw Strand	creates a single-stranded region of DNA
Simulate	turns physical simulation on and off

Some of these actions can be controlled with inputs that have already been seen in this text. The bonding interactions are implemented using the gun to select the pair of bases, one at a time, similar to the linking interaction seen in Chapter 5.

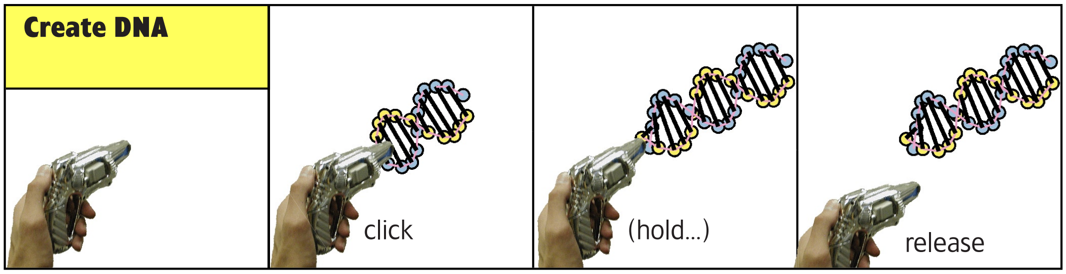




Simulate is a discrete action, containing only a mode transition. This is implemented with traditional methods (a 2d menu and a keystroke both toggle simulation state).

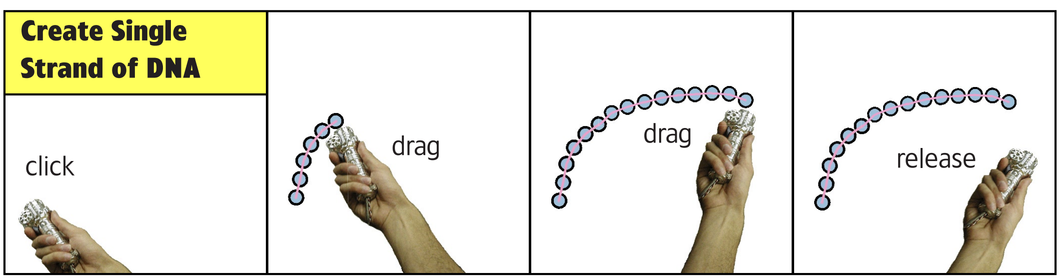
CREATING DNA

The appropriate interaction for placing DNA in space was a difficult design decision to make. One interaction that was considered was shooting the DNA out of the raygun.

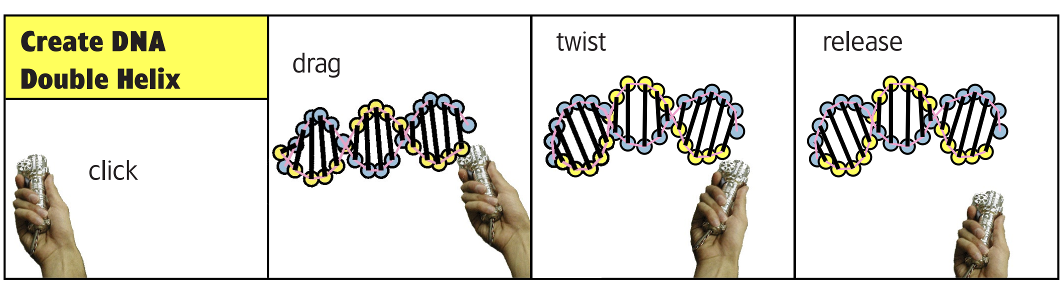


While this interaction offered sufficient control for double helices (which are rarely curved), it is less capable of placing a curved single-stranded region in space. Waving the gun in space to achieve the appropriate curvature does not map well to the kinesthetic frame of the gun. Moreover, this interaction does not control the beginning and end of the DNA molecule well, and a user must follow this action with a Reframe action to place the molecule in its appropriate spot. A device that emits the DNA molecule from some default position (such as out of the side of the screen) is no worse in this regard — as such the spatial data of the raygun is underutilized. 3d input devices are most useful, most efficient when their 3d coordinate frames directly relate to the action being specified.

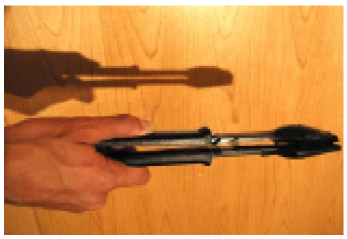
The handle places a single-stranded molecule with much more detail, tracing out its path in much the way it traced curves in Chapter 6.



When it comes to placing double-helices the curvature is much more limited (it takes 150 base pairs for a DNA helix to come around full circle [Sinden 1994]. For this reason, a different interaction is used where the starting and ending coordinate frames of the handle determine a DNA's curvature.



Proximity information, as shown in Chapter 5, can be used to change the tongs from Reframe-Scene to Reframe-Object mode. How can the tongs be used to Reframe components? One option is to place an additional switch in the tongs that detects when they are squeezed with great pressure.



Weak grab

The tong tips touch, but the middle does not (see shadow). This input triggers a Reframe-Object action if there is a molecule nearby, otherwise a Reframe-Scene is performed.

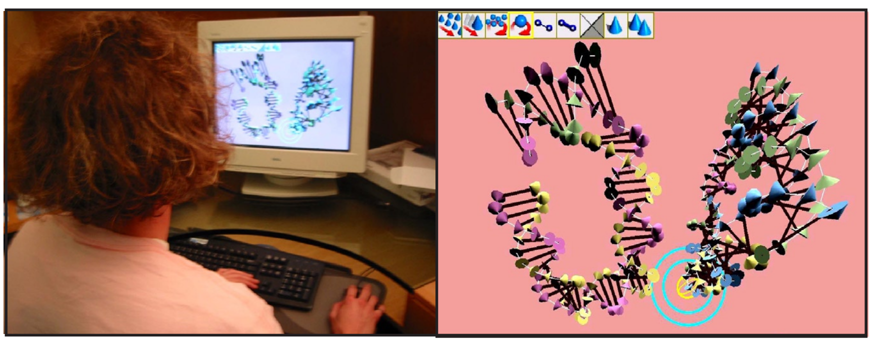
Strong grab

Note that the center of the tong is closed in the strong grab. If a molecule is close, this calls Reframe-Component, allowing the user to move one of the DNA bases.

A line is drawn to nearby objects and components to help discretize the interaction space. This approach is similar to the two-level buttons found on cameras, where a weak push enables autofocus and a firmer press of the button takes a picture. This method has been explored for mice and extended to 3d input devices [Zeleznik 2001 & 2002]. As the user study below shows, users had difficulty with two-level tongs for two reasons. They had difficulty with the fine motor control, sometimes switching from a strong to a weak grab in the midst of a Reframe. It is difficult to pay attention to nuances of grasp while one is focusing on a molecular manipulation. Another difficulty was due to the somewhat arbitrary association of a strong grab with Reframe-Component and a weak grab with Reframe-Object. This made sense in the initial design, when Reframe-Object actions were expected to be more casual and frequent, and Reframe-Component actions were thought to be more specialized and thus best associated with additional effort. Some users felt the opposite, assuming that since an object is larger than a component it is heavier and should be controlled with a strong grab. This difficulty led to a redesign of the Reframe interface which is described at the end of this chapter.

A 2D INTERFACE, FOR COMPARISON

This interface was compared against a standard 2d interface for manipulating molecules. No commercial tools could be found for the interactive construction of DNA molecules. In practice, molecular biologists draw design ideas with paper and pencil, making different sketches for the different levels of detail in a model. The sketches are transformed into 3d molecules with tools such as SpuriousC, which generates strings of base pairs that bind only where desired, and NAMOT, which allows the description of three-dimensional structure. NAMOT is far from visual, taking as its input a script in a formal language. When the shape is finally visualized, the design is many hours (even days) away from the original sketch model. Changes in the molecule require iteration of the entire design process. The inadequacy of these tools, and the difficulty of manipulating DNA with traditional interface techniques, in part motivated this project. Because no existing 2d interfaces for DNA construction were available, I built an interface to the same underlying DNA construction software using the mouse and 2d monitor.



The camera control in this interface was the same as that used in the 2d interface for arrangement seen in Chapter 5. Clicking on icons at the top of the 2d screen mapped the mouse pointer to the following functions:

Translate molecule	Clicking on a base illuminates the translation widget seen in Chapter 3: a square around the base in the screen plane, and three coordinate axes in the base's local coordinate system. Clicking on the center square and moving the mouse translates the molecule in the image plane. This is analogous to a weak grab with the tongs (note that the tongs allow rotation and translation to occur at the same time).

Translate base	Similar to translate molecule, but this time only the selected base is moved. This is analogous to a strong grab with the tongs.
Rotate molecule	Clicking on a base displays a local arcball widget (as illustrated in Chapter 3) around the base's center. Clicking and dragging rotates the molecule.
Rotate base	Similar to rotate molecule, this tool only affects a single base.
H-bond	Clicking on base 1, dragging the mouse, and releasing it on base 2 forms a Hydrogen bond between base 1 and base 2.
P-bond	Same as above, creating a Phosphate bond.
Draw Helix	Clicking, dragging, and releasing draws a double-helical region between the endpoints in the image plane.
Draw Strand	Clicking and dragging draws a path in the 2d screen plane consisting of linked bases, the orientation of which is specified by the direction of mouse movement.

A method for toggling the simulation mode was not included in the 2d interface, and simulation was active for the duration of the user study below.

USER STUDY

The subjects are six research scientists (PhD students and postdocs) who study (or studied) DNA, one of whom is female. We asked them to build several molecules with both interfaces (which we referred to as 2d and 3d) and then fill out a questionnaire describing their experience. The scientists were taught each interface (half the users started with the 2d task, the others started with the 3d task) and demonstrated an understanding before proceeding with five timed trials. We alternately started with the 2d and 3d interfaces. Each interaction was described, the subjects demonstrated their understanding of each tool, and we ran five timed trials. In these trials, the subjects attempted to draw a DNA hairpin, a Holliday junction, and a DNA cube. A Holliday junction consists of two aligned helices whose strands cross from one helix to the other. The DNA cube is significantly more complex — each edge of the cube is a double helix, each face has one continuous piece of DNA circling it, and at each corner the three intersecting helices swap strands with one another. Thus the tasks include both simple molecules, and a very complex one (the cube) that would be virtually impossible to complete in the five minutes we allowed for each task. The deliberately difficult tasks pushed the subjects to the limit of their ability to manipulate space in each interface.

On an emotional, intuitive level, all of the subjects preferred the 3d interface — many displayed great enthusiasm. For example, one user, being told his time for the 3d task was up, complained "oh, but I'm having so much fun!" This comfort comes despite extensive familiarity with 2d mouse-based interfaces. (Only two of our subjects had used immersive interfaces, and those only very briefly). The primary strength of the experimental 3d interface seemed to be the natural rotation and placement of objects in space that direct union enables:

> *The 3d interface was less interference between me and the molecule. Working in the 2d interface, I was spending my time figuring out how to position the space so that I could access the relevant parts of the molecule with the 2d tools. That is not the kind of creative thinking I want to be doing!*
>
> *Looking at different parts of the molecule by moving my head was very natural. It felt like there was no "interface" at all. Rotating and/or moving the space or molecules with a single pair of tongs was very natural.* [both quotes from user #1]

In contrast, the users found spatial management quite difficult with the 2d interface:

> *I had trouble rotating things and understanding what was closer to me and what was farther away. Also, I didn't really know what I was doing with the rotation except when I was rotating about the axis normal to the screen.* [#2]

This difficulty seems to stem from the inherent difficulty in mapping six degrees of freedom to a 2d mouse. Our subjects did not find many strengths with the 2d interface. Many subjects said that drawing helices was the best of the 2d interactions. But those same subjects also said that drawing helices was easy in the 3d task. Others cited familiarity and portability as advantages. Cutting was found to be easier in 2d:

> *Cutting a bond was easier for me in 2d than in 3d, perhaps because I knew I had to position everything so that the bond was clearly visible and distinct from surrounding clutter, so when I was ready to cut it, it was easy to cut.*
>
> *Cutting bonds with the sword, I had to concentrate to make sure I would be cutting the right thing. I would have to make sure my point of view gave me a good view of the entire blade.* [both quotes from user #1]

These observations indicate that there is some strength to a 2d interface where the world is not continuously moving, as it is in 3d space due to head-tracking noise. We feel that this crispness is the primary strength of the 2d GUI. The second quote from the user above indicates that sometimes direct union is actually not advantageous. In this highly cluttered scene, cutting from a distance is preferred by this user because it can be executed predictably. Contradicting this opinion, another user preferred the 3d interface for cutting, valuing the high bandwidth that 3d interaction enables:

> *Cutting bonds precisely with the lightsaber tool is easier because the plane of rotation of the cutting edge can be changed.* [#6]

The users were not faster at the experimental system, nor did they produce more complete designs. Some users found the experimental system superior to pencil and paper for sketching out ideas, while saying that the 2d system would be best used in addition to pencil and paper:

> *When using the 2d interface, I wished I had pencil and paper so that I could sit and sketch things, and make a plan of attack. I never thought this with the 3d interface. When using the 3d interface I immediately saw things that would be very difficult to put on paper, and I felt that the interface was a very natural tool for trying things out.* [#2]

> *The 2d tool didn't seem like a big improvement over pencil and paper, even though it was representing a 3d model. It might still be useful, but it was kind of a hassle to use, so I'm not currently inclined to use it.* [#1]

The greatest difficulties with 3d were accidentally triggering a strong grab with the tongs when a weak one was attempted:

> *I had trouble with the "weak" vs. "strong" usage of the tongs.* [#3]

> *The distinction with the tongs between moving a single atom or an object should be made crisper, the squeeziness of the tongs is a little subtle (but I like it actually). [#6]*

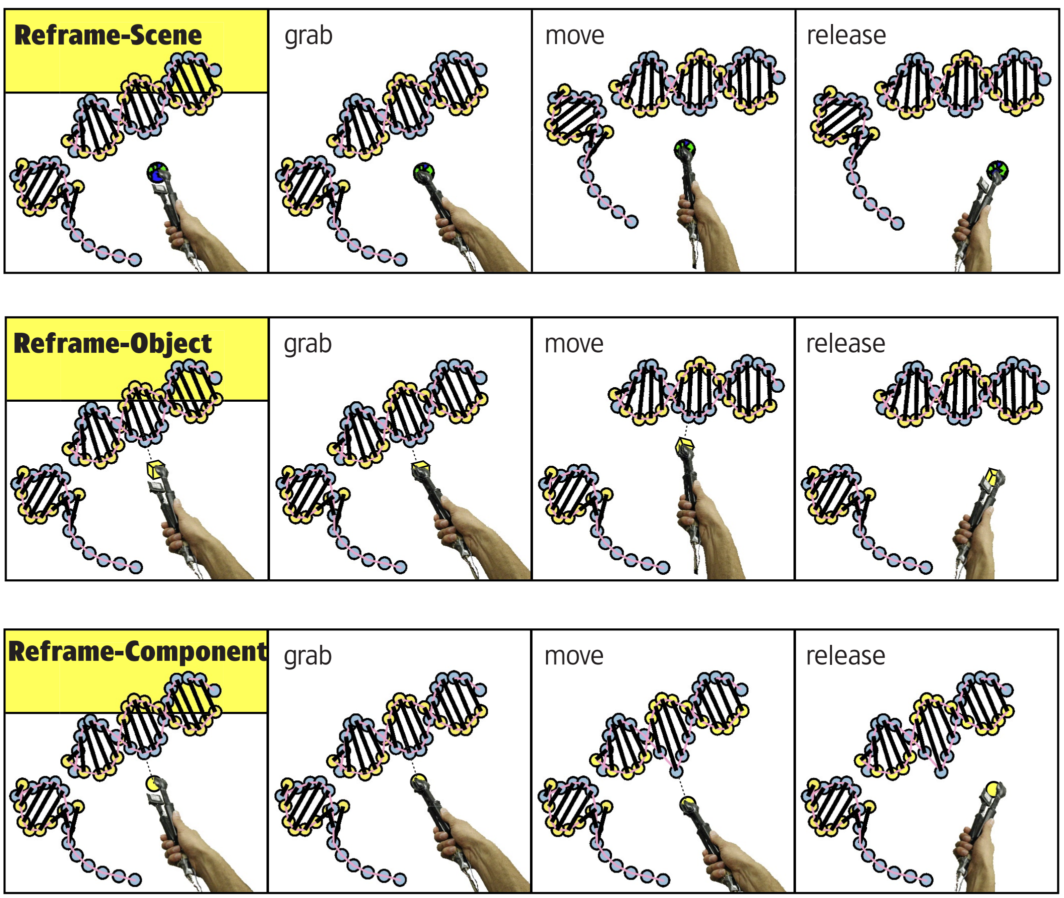
These users are having difficulty with the continuous mode selection inherent in the weak vs. strong distinction. This problem would be eased by either making the difference between the two positions more distinct, or by selecting a different metaphor that did not rely on continuous mode selection. Specific aspects of the interface aside, the gestalt of the 3d tangible interface aided these researchers in the intuitive manipulation of structure that is crucial for scientific insight. When asked which interface better supported creative thinking and spatial manipulation, the subjects responded:

> *The 3d interface: it gives me a much more accurate picture of what is really happening. I don't waste time thinking about geometric misunderstandings, and can really think about what I am building.* [#2]
>
> *Certainly the 3d — I can more completely see the medium with which I'm working.* [#3]
>
> *3d! It seems more natural, you don't need to remember which keys are which (though with time, it might not matter), but it's helpful to be able to "grab" something just like you would in reality.* [#5]
>
> *The 3d interface, without question. By just glancing at the image, I have a better understanding of the structure. But the value of the 3d interface is much more than just nice rendering. By being able to intuitively manipulate the structures I could have a manual understanding that augments the visual understanding. I usually think about 3d objects with my hands, and this interface suited me very well.* [#2]

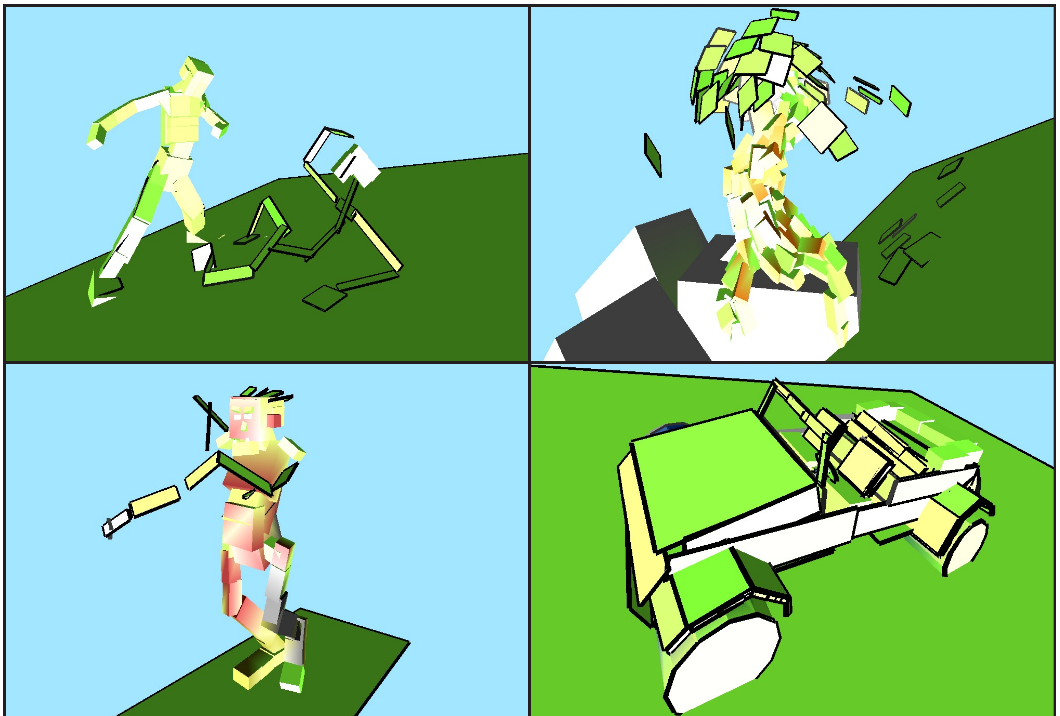
While running the study, the difference between agitation using the 2d interface and enthusiasm for the experimental design was apparent. As user #4 simply wrote: "3d is more fun." The connection between enthusiasm, attention, and insight suggests that 3d tangible interfaces could have a crucial role in technological innovation. It is important to note that we did not see a qualitative difference between the molecules designed in the two systems. While all users managed to build successful hairpin molecules within the allotted time, very few finished the Holliday junction, and none finished the cube. For these more complex designs, the users spent much time building strategies for construction. We feel that differences in design quality would emerge only after more experience.

EXPLICIT TONG MODES

Another phrasing of tong input addresses the difficulties the users encountered with weak and strong tong grabs. Instead of proximity and strength of grasp being used to specify a reframe level, a separate accessory explicitly indicates when Reframe-Object or Reframe-Component will occur. This further discretizes the input, making it even more clear what will happen when the tongs are closed, at the cost of the additional interaction necessary to change the tongs from Reframe-Object to Reframe-Scene to Reframe-Component mode. The strong grab triggers a halo menu with three different accessories. Lines from icons to target objects help users see which object will be selected when the tongs are closed in Reframe-Object and -Component modes. In contrast to the method of choosing Reframe level based on the distance between the tool and the digital target, this method was designed so that users could discretely choose which Reframe they desired. In practice, both for sample users and for my own experienced hand, this greatly reduced selection errors. The three variants of Reframe are shown below:



This interaction was favored by an artist who worked with the experimental system for some two months. He used the Drop action, these Reframe interactions, and the ability to reframe accessories to create the following structures out of simple geometric primitives. The shapes were dropped with the handle, which had eight different primitives on its halo menu.



The characters, tree, and Jeep seen above were created by placing and manipulating simple primitives using a variety of accessories mapped to a few tangible tools.

8 materials and methods
building input devices

BUILDING 3D INPUT DEVICES and creating 3d displays is non-trivial. 3d interfaces are somewhat a black art in this respect, there is no standard way they are used and the interactive system designer is left building components, and sometimes retrofitting tools that are designed for other applications to realize their interfaces.

In the experimental interfaces discussed in this text, electromagnetic trackers and MIDI sensors were used in combination with passive and active rear-projected stereo to create the 3d experience. These interactions could be rendered equally successful with different underlying technologies. This chapter describes some of the options available to the interface designer for display and sensing.

DEPTH PERCEPTION

The main purpose of 3d display is to establish a more direct spatial relationship between body and the digital space than possible with flat 2d screens. There are a variety of cues that psychophysicists believe create the visual perception of 3d space:

- shading (lighting cues in the environment)

- occlusion (objects further back in space are partially or completely hidden from view)

- texture frequency (along textured objects angled away from the viewer, the frequency of a texture's projected image on the retina increases as the object recedes into the distance)

- perspective cues (the size of an object's projection on the retina changes as it moves towards / away from the viewer)

• motion parallax (when the head moves, the projected images of objects on the retina move at different rates depending on distance from the eye)

• binocular disparity (the projected image of an object in the retina is different for different eyes, allowing a depth measure to be computed in the cortex via a kind of triangulation)

• convergence (as a point moves closer to (or further from) the viewer, the angle between the eyes focusing on that point change)

Of these cues, note that the first four are mimicked by standard 2d monitors displaying scenes using established 3d rendering algorithms. Advanced 3d display techniques provide viewers with additional cues which enhance the sense of depth. Foremost amongst these is binocular disparity, displays which provide this cue are called stereoscopic. Convergence is still difficult to replicate with computer displays, although particle-based displays are a promising possibility for true volumetric imaging.

STEREOSCOPIC DISPLAY TECHNIQUES

There are four established classes of stereoscopic display: holography, head-mounted displays, glasses-based stereo screens, and autostereoscopic display screens that do not require glasses. The earliest stereo computer displays were head-mounted displays such as that developed by Sutherland in the 1960's [Sutherland 1968]. Anaglyph (red-blue) stereo displays were common in the motion picture industry a decade earlier. This trend was strongest in the early 1950s, but did not find lasting success, ostensibly because 3d viewing did not significantly enhance the movie experience.

Head-mounted stereo displays place screens near the viewer's eye, with one display surface per eye. These systems are ideal for the presentation of a virtual reality that separates the viewer completely from their physical environment. The major limitations of this method are the weight of the display, the inconvenience of wearing a display, and the difficulty of providing a high-resolution display with a wide viewing angle.

Display manufacturers are making steady progress on these issues, although a head-mounted display with the 60 degree field of view necessary for a sense of immersion are still bulky and expensive (upwards of $10,000). Head-mounted displays are particularly inappropriate for the input methodology developed in this text because they separate the viewer from the physical world, and detach users from their physical input devices. Video see-through displays are a head-mounted display technique where a video signal of what the user sees is composited in realtime with a digital representation, reducing this detachment. Once occlusion issues are solved and realtime low-latency compositing algorithms are improved, video see-through displays offer a quite attractive display setup for spatial construction. One of the chief advantages of head-mounted displays is that they are physically small. The other display techniques mentioned in this chapter require a quite large screen to accommodate the user's body space, and those based on projection moreover require a large space for the adequate projector throw-distance to be established.

Passive and active stereo are two approaches that use glasses and filtered screens to provide the stereo effect. Active stereo refers to the use of shutterglasses, whose lenses are LCD screens that can be made to appear either black or transparent. These glasses flicker quickly, alternating which eye is blocked and which eye can see the screen.

These displays require fast projectors, about 96 Hz (48 frames per second per eye) to provide the impression of a continuous view in each eye. Most off-the-shelf projectors do not switch this quickly, and most off-the-shelf monitors are not large enough for full-body interaction. Another option is passive stereo, where two projectors beam overlapping images on the same screen. Filters polarize the light, and the viewer wears a pair of polarized sunglasses to decouple this filter, so that each eye sees only a single projector's image. There are two varieties of polarization. Linear polarization (horizontal and vertical) typically yields a stronger separation, but if a viewer rotates their head they lose this separation. Circular polarization (clockwise and counterclockwise) is not sensitive to the head's angle, although its separation is less strong than that found in linear filters resulting in ghosting (a faint right-eye image appears in the left eye and vice versa). These are the techniques that were used in the interfaces in this text.

Autostereoscopic displays present a stereo view without glasses. This is typically done with a moving parallax barrier which slides over the screen, allowing each eye to see selected columns of the screen. The chief limitation of this method is that it requires a precise knowledge of where the eyes are, otherwise the image is lost. However, for fixed viewing autostereoscopic displays, which are rapidly coming to prominence on the marketplace, are quite attractive.

Another important depth cue is motion parallax — as the head moves, does the 3d image respond by shifting the placement of objects in relation to the viewer? This is typically achieved by head tracking: aligning the virtual camera used to render the scene with the position of the user's eyes. The various methods for sensing the head and other physical objects in the interactive environment is the subject of the next section. For the meantime, let us first say two things about head tracking. First, it is commonly held wisdom in the virtual reality community that head tracking is an essential component of compelling immersive experience. Second, my recent experiments in an environment where the user is seated in front of a stereo screen display little difference in the quality of experience when head tracking is turned off. One of our artists became annoyed at wearing a head-tracker and started placing it on his shoulder instead of his head. Apparently precision in this regard is not necessary for modeling, and it is simpler not to use a device. When the user's head moves very little, the extra bulk of a head tracking device, and the jitter that it lends to the scene due to latency between tracker and visual update offset the benefits of a perfectly perspective-correct view when the user settles in their new position.

SENSING

There are a host of methods to sense physical objects so as to create alignment between physical space and visual display space. The most common parameters to sense are a coordinate frame, commonly referred to as 6DOF (six degree of freedom = three position variables and three orientation variables) tracking. Some technologies sense absolute positions and orientations in space, and others (such as accelerometers) only sense differential tracking. Sometimes a high-update rate incremental tracker is combined with a low-update rate absolute tracker [You 1999]. Electromagnetic trackers emit a magnetic field from a fixed point in the interactive space which induces a current along wires in receivers which are either mounted on the user's body or built into input devices.

An outstanding issue with magnetic trackers are wires, there are currently no completely wireless magnetic tracking systems on the market (one partially wireless system built for motion capture runs wires from the user's body to a back-mounted wireless transmitter). Optical tracking systems can be built without wires, in these systems cameras sense completely passive fiducials. These electronics-free markers have no need for wires. These systems find performance roughly equivalent to magnetic solutions, although the environment must be carefully designed so that there is always a clear line of sight between the fiducials and two or more cameras in the room. Other optical approaches, notably the Hi-Ball tracker, achieve very high absolute position tracking by mounting a camera on the user (or handheld input device) which uses structured light (ceiling-mounted LEDs) to derive the position of the tracker. Again care must be taken to not to block this camera during interaction, and to sufficiently surround the user in structured light.

part III:

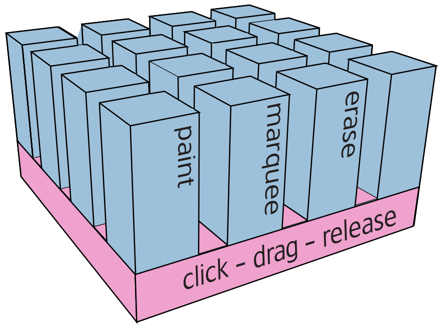
interface theory

9 balancing specificity

supporting cultural affordance and kinesthetic framing without sacrificing generality

This text presents tangible controllers for actions in 3d space. While tangible interface's traditional strength is physically uniting humans with data, the chapters in Part II demonstrate functional advantages of TUI: kinesthetic framing and cultural affordance. Enhancing cultural affordance and kinesthetic framing often results in an impractical interface that is over-specialized. Users don't want to buy a special interface for each task. A special interface must be set up before it is used, it is difficult for software developers to predict what special devices a user might have, and in some cases specialization increases the amount of learning users have to go through before performing a task. For the remainder of this chapter we touch upon these issues of kinesthetics and culture, but moreover we focus on a widespread issue, and that is the tension between generic and specific input devices.

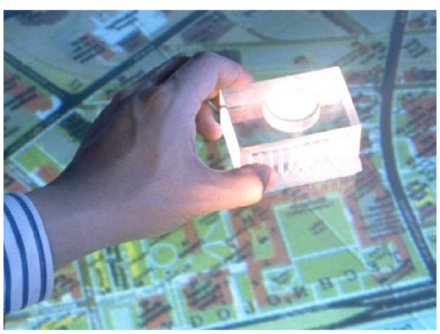
The standardized 2d pointer, controlled by a mouse, trackball, or pen, is quite generic. This single metaphor is mapped to many digital actions: selecting text, selecting files, zooming into a region, painting strokes, rotating objects, etc. Not only does the form factor of the input device remain constant across these operations, but moreover so does the phrasing of input — in all of these examples the mouse is clicked, dragged, and released to perform an operation. We can visualize an interface as a small city — each building is an action which is supported by one of the available inputs. In the case of the mouse, every building is supported by the same foundation.



Highly generic input

The mouse is mapped to a vast array of operations, many of which are activated with the same click, drag, release metaphor.

A quite different approach is to have one input device for each kind of functionality in a system. A good example of a complete system built with this philosophy is the metaDesk [Ullmer 1997]. The Tangible Geospace application on the metaDesk allows users to navigate a map of MIT campus. There are several devices that allow different operations such as a special tool to rotate and scale the map, an arm-mounted LCD display that shows further detail, and the Great Dome phicon.

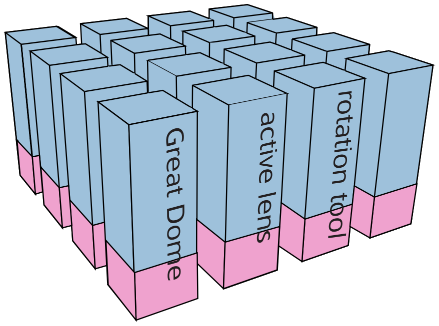


The Great Dome phicon

A phicon linked to a unique place on a map. The digital map always places itself so that the location of the Great Dome is underneath the physical icon [Ullmer 1997].

This interaction is the quintessence of specific interface, in that not only does the phicon perform a single function, it also is tied only to one map — on a map of anything but MIT campus, this metaphor is inappropriate. Note how strong the cultural tie between input and action is in this case — the Great Dome exhibits a great degree of cultural affordance in its identity, although it does not necessarily present a large amount of cultural affordance in its use (it is not typical to move maps with buildings).

Continuing our building metaphor, a very specific interface is a city whose interactions as not unified by common input methods.

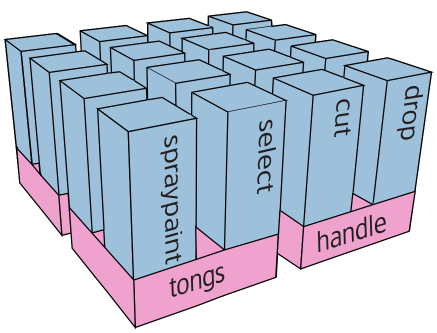


Highly specific input

In the metaDesk interface [Ullmer 1997], each device is mapped to a single function. In this case input and action are merged — throughout the interface they are always coupled.

As such we see a number of the weaknesses of specific interface. Not only does the user need to learn and remember many different inputs, they also need to own and manage many input devices. This highly specific approach also makes it difficult for the flexibility of software to be employed to change the function of computers incrementally and on the fly. The advantage of this approach is the ability to customize: there is a tight mapping between form and function and the opportunity to give each device a heavy dose of cultural affordance.

My philosophy is to strike a balance between highly generic and task-specific interactions. The city described in Part II is built on a foundation of four input devices.



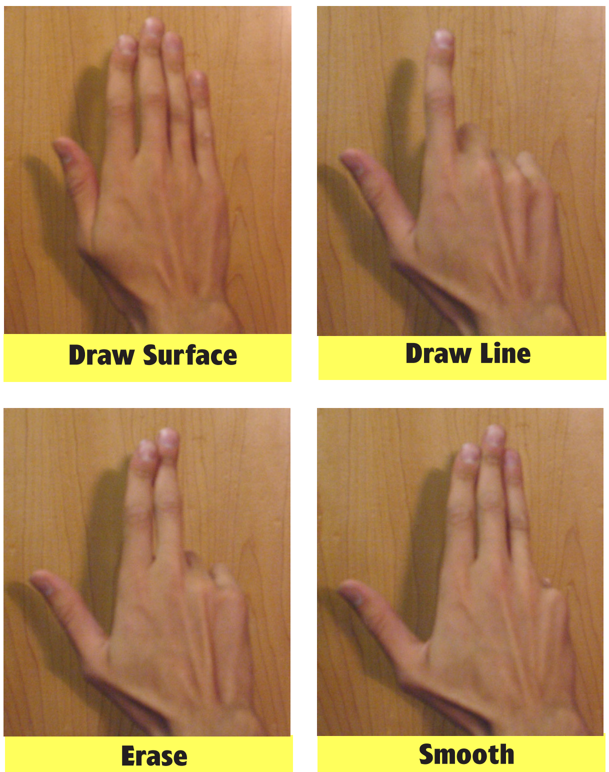
A balanced approach

The tongs, raygun, handle, and hand segregate the action space into four regions, each of which are controlled with the most suitable input device.

Clustering simplifies interaction and reduces the number of devices, while still allowing a reasonable amount of customization of each device to its task. Their cultural significance is linked, not to data as in the Great Dome example, but rather to functionality. A culture of use can be applied to many different situations. In addition, each of these devices has a unique kinesthetic framing — the surrounding space is framed by the device itself, mentally informing a user's actions as well as physically easing the act itself. Before I elaborate upon the kinesthetics of these devices, let's consider a generic approach to spatial construction that not only displays scant cultural affordance, but also shows how **not** to establish a strong kinesthetic frame for 3d interaction.

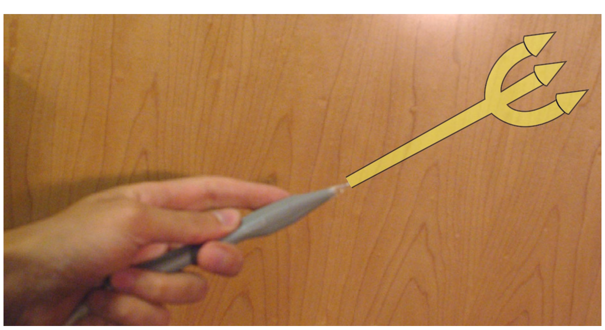
HAND POSTURES FOR INPUT

An early interface for freeform surface creation [Schkolne 1999] used hand postures to distinguish between tool states, and ran into some successes and many difficulties along the way. One hand posture controlled each of four tool modes:



Hand postures from an early surface drawing interface that mapped each posture to a different action.

These postures were differentiated using the continuous angle sensors found in the CyberGlove, which the users wore for the duration of the interaction. In addition, a stylus with a fork icon was used to move the whole scene.



A fork accessory

For the early surface drawing interface, a stylus with a fork accessory was used to move the scene. Objects could not be moved individually, only Reframe-Scene was available.

We showed this interface to about 1000 users at SIGGRAPH Emerging Technologies in 1999. The feedback from the users was characterized by successes and shortcomings.

- Users liked drawing surfaces with their hand. Many artists said they felt connected to computers in a way that they had not experienced previously — some artists added that they hadn't found a way to use computers in their work, and they hoped this type of technology would soon become available to them. In particular, they appreciated not having to think mathematically to create 3d digital shapes. They favored the material representation to the symbolic methods found in much computer modeling software.

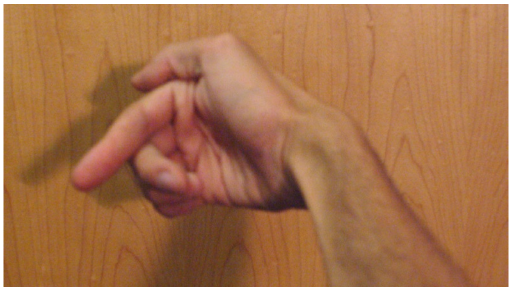
- Users found the four hand postures to be **arbitrary** — they often had to be reminded several times during a session that two fingers meant erase, and one meant draw a line, etc.

- Users found the hand postures to be uncomfortable — some users visibly struggled to make the poses, and some users (for example, an older participant with arthritis) could not make them at all.

- Users were delighted by the trident-like appearance of the fork that was attached to the stylus. They found this cute and inviting. While they weren't immediately sure what it did, it was very quick to explain it to them.

- Users had a great amount of difficulty with the thumb switch. At times, while making a stroke they would accidentally open the thumb. Conversely, they would sometimes accidentally close the thumb when they didn't mean to. This is further evidence that the mismatch between **continuous** thumb angle and a **discrete** on/off state transition is fundamentally flawed. Both this difficulty, and the one noted below, were furthered by the nature of a glove's sensors, which move over the surface of the hand as it is used.

- Users had difficulty maintaining a hand posture for the duration of an interaction. While at first glance the postures seem clearly delineated, in practice the hand twists, turns, and bends as the arm is moved through space and around a shape. The image below shows an intermediate posture, somewhere between draw-surface and draw-line, that many users unintentionally made. (The reader is encouraged to move their hand through space and curve their fingers to better understand how this posture is encountered). If you are surprised that this happens, so was I — this difficulty was not anticipated in the design process.



A posture in the middle

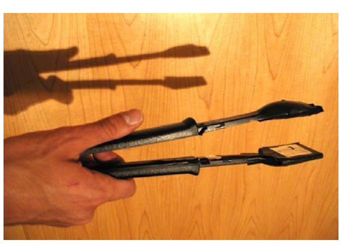
An inbetween posture that users often accidentally slipped into while drawing — switching from drawing lines to surfaces or vice versa. When sweeping the arm through space, it is difficult to closely monitor hand posture.

While this hand-based input satisfied a long-standing goal [Krueger 1985] of using the body itself as input, this interface does so at the expense of user control, comfort, and intuition.

SPECIALIZATION WITHOUT LOSS OF GENERALITY

It is difficult to map the hand to a range of tasks. Perhaps the hand in it's naked form is not ideal for controlling a diverse set of actions? Amongst creative tasks, there is abundant use of hand-held tools, especially when precision of some sort is required. Consider someone making coffee with an espresso machine — there are a variety of vessels used to control the coffee, not just because of its heat, but moreover due to the fluidity of the constructive medium (coffee). Tools provide valuable assistance for spatial construction, not only due to their ability to hold fluid digital structures, but moreover because they make input less arbitrary, integrate input devices into a deeper culture of tool use, and provide subtle cues that guide and constrain interactions.

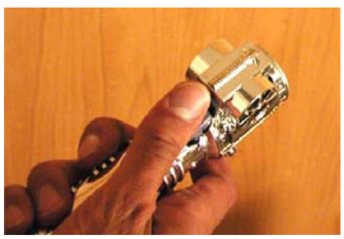
Unlike the hand mappings, the tools presented in this thesis display a clear **cultural affordance**: when a user encounters the tool, they have some inkling of how it will function. This history has a minor role in improving task function, but a major role in making a user emotionally at ease with an interaction. Cultural affordance provides a level of familiarity, a natural feeling rooted in a collective consciousness that goes back centuries. One of the strengths of this approach is the deliberate relationship between form and function.



Tongs have a long history of use for grabbing objects. While the average user might know these tools best from the kitchen or barbecue, similar devices have been used to work iron for centuries.

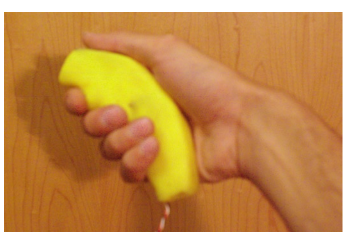
The **raygun** is modeled after firearms, which also have a long tradition of use, primarily for hitting distant targets. While the applications of the raygun are non-violent, it similarly hits small objects from a distance.

The surface of the **hand** is used for smoothing objects, culturally rooted both in the act of polishing stone with a light cloth and smoothing loose earth such as dirt or sand. The hand is also used to trace forms, a method regularly employed in conversation to describe shapes.



The **handle** is rooted in devices such as swords, tennis rackets, bicycle handlebars, suitcase handles, torches, and so on. Many devices use this base to control a more complex end effector. The handle is used to control tools, such as the lightsaber, and place a wide variety of objects.

The shape of the computer mouse, in comparison, is incredibly non-specific. Original mice were small boxes, and more recent models are shaped like vague ergonomic blobs. As such the mouse's form is somewhat arbitrary, unrelated to cultural precedents. A similar design fork was seen in the construction of the handle, an original form took the shape of an amorphous blob.

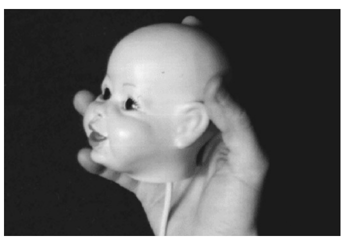


An original version of the handle tool designed, not to suggest meaning, but rather for ergonomic support. This form has a much weaker cultural connection than the handle shown above.

This form, designed to fit the hand ergonomically, frequently confused users (who often held it incorrectly or couldn't identify the button). The silver handle that was built from a toy lightsaber's handle was less confusing: users always knew how to use it, with no sacrifice in ergonomics. The distinction between these blobby forms and culturally rooted ones lies in their distinct, deliberate form.

deliberate	**arbitrary**
clear link between form and function	form unrelated to function
necessary mapping between action and input	input unrelated to action
easy to remember	learning required
user comfort, immediacy	occasional user confusion
cultural affordance in input device form	amorphous input devices

Alongside the Great Dome phicon, the doll's head prop used by Hinckley, Pausch, Goble, and Kassel for neurosurgical visualization is both quite deliberate and rich in cultural connotation.

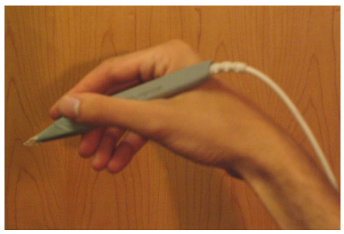


One of the few prior examples of 3d tangible input devices, a doll's head is used by neurosurgeons to control the placement of a model of a patient's brain in a visualization application [Hinckley 1994].

These prior methods make data tangible. The methods in this text make functionality tangible. In the prior work, the shape of the tool is used to form a strong connection with a piece of data: *the tool is a physical handle for data*. Hinckley's doll's head is a physical proxy for a patient's skull. In the current work, the shape of the tool informs and guides action: *the tool connects body to function*.

The current weakness of relationship between tool form and tool action in space can be scene with the current generic device heavily used in the virtual reality community: the sensed stylus. This tool was used to move objects in the first interface described in this chapter (alongside the hand postures). This tool's form is inconsistent with its use. The stylus is designed for fine motions, such as writing (compare with a pencil). In our experimental application, where the stylus moves large objects, users in the exhibit frequently held the stylus in what is called the *power position* [Napier 1956, Ehrsson 2000].

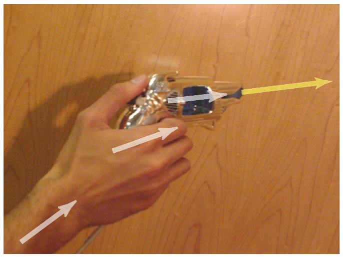
In a **power grasp**, all four fingers are used to control a tool with muscles and maximum applied force..



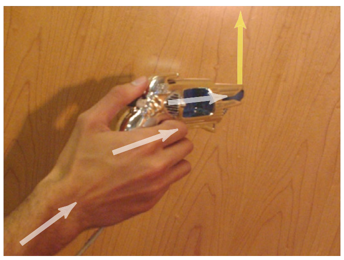
In a **precision grasp**, the thumb and two fingers are used for fine scale manipulation.

The power grasp is naturally supported by the handle form. In body-scale spatial construction, this type of motion which uses the elbow and arm is more frequent than precise motions using the fingers and wrist. This emphasis on strong, power moves is one of the reasons that high precision tracking is not necessary in my interfaces — if precision is required, users scale the scene until the precision matches that offered by the trackers. This change in grip raises a larger issue of **kinesthetic framing**. This concept refers to the relationship between the shape of a device and the structure of interactive space around the tool itself. Kinesthetic framing has antecedents in Mine, Brooks, and Sequin's analysis of proprioception in virtual reality spaces [Mine 1997] and Balakrishnan and Hinckley's treatment of the kinesthetics of two-handed input [Balakrishnan 1999] (which in turn dates from Guiard's seminal work [Guiard 1987]). Many natural devices have clear kinesthetic frames defined by their physics. A screwdriver has an affordance that allows it both to be rotated easily around a central axis, and for force to be delivered into the material being affixed by the screw. The majority of a screwdriver's form serves to provide this kinesthetic frame (only a small portion of the device touches the screw itself).

Similarly, the 3d tools have properties of form which facilitate their use for certain classes of action in digital space. Some of the kinesthetic frames are relationships which live very close to the transition space between physical form and digital accessory.

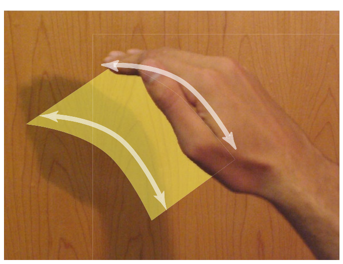


Kinesthetic match: The beam of the gun is in the same direction as the arm and the index finger, which makes the action similar to pointing. In addition, the barrel of the gun physically points in the same direction as the digital beam.

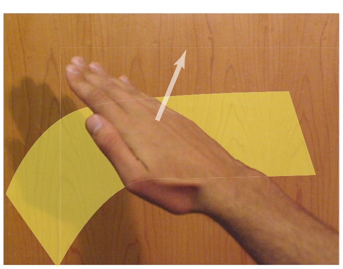


Kinesthetic mismatch: The beam of the gun is perpendicular to the dominant direction of the arm, finger, and raygun. There is a psychological and physical mismatch between the tool and its effects in this example.

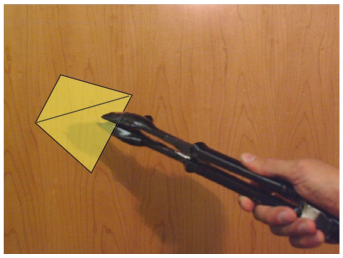
Kinesthetic support: The handle is designed to facilitate the power grasp instead of the precision grasp. This grip is more appropriate to an environment where tracking is imprecise. One way of interacting effectively with low-precision trackers is to use power grasps in large-scale worlds.



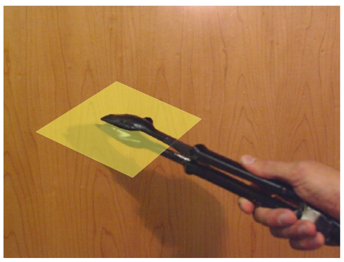
Kinesthetic mimicry: The curvature of the hand is matched in both the accessory the hand controls (an iconic abstraction of the finger) and the stroke that is formed. This establishes a correspondence between kinesthetic space and action space, uniting the user's inner proprioception with their creation.



Matching frames: In the smoothing interaction, the normal vector of the hand is used to guide the final slope of the surface. In a sense, the user is telling the system to have the digital surface match the hand's surface: there is a tight connection between the joint positions in the hand and the digital product.

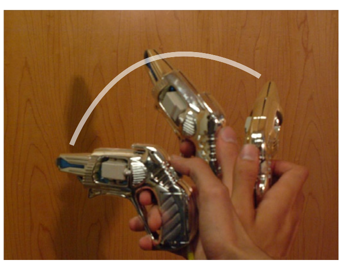


Kinesthetic reference: The closing of the tongs refers to the act of closing around a physical object. While there is not the resolution of force feedback that would be present if the display was fully physical, this sense enforces the act of grabbing in the user's mind.

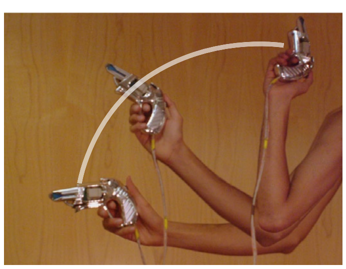


Kinesthetic reference: A higher-precision kinesthetic framing takes into account the angle of the shape being grabbed and only allows grabs in certain circumstances. This was not implemented in the experimental interfaces because it requires much higher user attention.

Another aspect of kinesthetics has less to do with the immediate relationship between tool and form, and more to do with the larger motions of digital objects. These kinesthetic relationships live farther from the tool/hand intersection. For example the raygun can sweep a large volume of space by rotating the wrist or moving the arm.

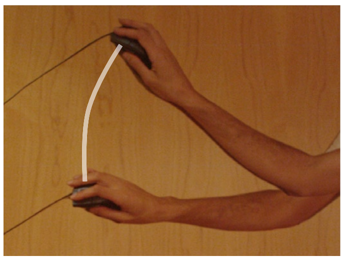


Rotating the wrist sweeps the beam of the raygun through a cone of space.

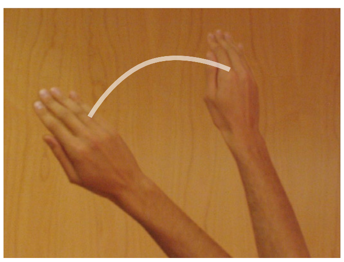


Larger motions of the arm, using the elbow in particular, sweep the raygun through a wider range.

Extending the arm increases the accuracy of the raygun, making up for some of the orientation jitter present in the trackers. Like the example above, this would not be as facile if the gun's beam emitted from the top of the gun (as in the kinesthetic mismatch above.)

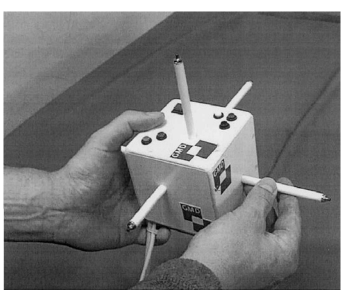


The mouse also exhibits a large-scale kinesthetic match between the elbow and the up/down motion of the controller, similar to the kinesthetics of the raygun shown above. In addition, moving the mouse to the left and right is accomplished with a rotation of the forearm around the elbow.



The hand tracing intearction leverages a similar rotation of the forearm to sweep space. This lateral motion of the hand Is tied to forearm rotation, which is not as clearly linked to joint angles as seen above. This motion is comfortable (the most natural for rubbing, for example) but not as clearly tied to kinesthetics.

The small and large-scale kinesthetic relationships illustrated above affect the ability to function in space. The Cubic Mouse [Fröhlich 2000] is a device which clearly links its form to data.



The cubic mouse

This device is designed to manipulate cutting planes in medical visualization application. Each bar controls the placement of one of three orthogonal planes. There is a direct relationship between a single bar and a single cutting plane.

The Cubic Mouse, like much research on immersion and virtual reality, focuses on data manipulation, and thus a link between data and form is key. In my research on spatial construction, function (changing data) is the focus, and the kinesthetics of action are a primary concern.

The handle and the tongs leverage a large-scale kinesthetic relationship which is different from that of the raygun and hand. A dominant direction is not present. In these situations control of all six degrees of freedom is maximized. The concept of direct union, which is discussed in detail in the next chapter, touches upon the strengths inherent in explicit representations of tool position in digital space.

RECONSIDERING THE HAND

Can a single device be used for all four classes of action we have seen? In line with this chapter's theme of specific vs. generic interface, we close with a thought experiment: can we apply these kinesthetic and cultural concepts to deviceless input with the hand? Users repeatedly ask for the ability to simply interact with the hands in 3d space. There is such powerful common sense to this notion, and such a demand from users who constantly mention it, that I am led to consider this option despite the sensing difficulties we have seen. My solution below leverages the advantages of the physical tools that this thesis introduces. The primary difficulty in this experiment is the disjunction between the continuous input space of hand posture, and the discrete actions of swapping hand functions. An ideal mapping of the hand to actions in 3d space exhibits the following properties:

- four hand modes:
  - grab (replacing tongs)
  - hold (replacing handle, also covers one dimensional tracing)
  - point (replacing raygun)
  - rub/trace (the 2d or higher version, with continuous finger modulation, and a tangent plane related to the hand's surface)
- discrete inputs activate a switch between hand modes. in particular, we would like to:
  - switch between hand modes,
  - activate a tool, and
  - activate a local menu to remap a hand mode to a different accessory

  all with discrete input.

• full support for four classes of accessories. users can swap between a host of accessories for each hand mode. Ideally there are separate local menus for each hand mode to minimize menu clutter.

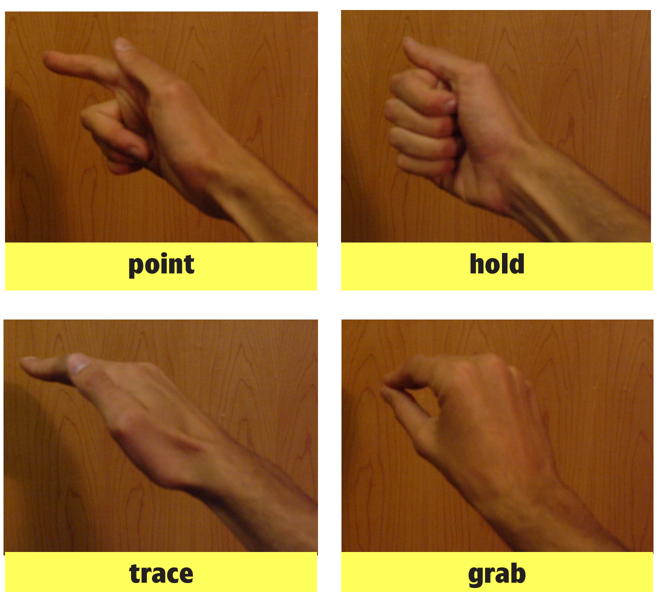
• appropriate kinesthetic frames for each hand mode

• accessory placement should make sense relative to hand placement in all modes

• all hand postures should be comfortable, especially those that are held for a prolonged period of time. this criterion suggests that only major muscles in the hand should be used for inputs, and that no postures should require a high degree of hand flexibility.

• each posture should be close to a culturally established hand motion

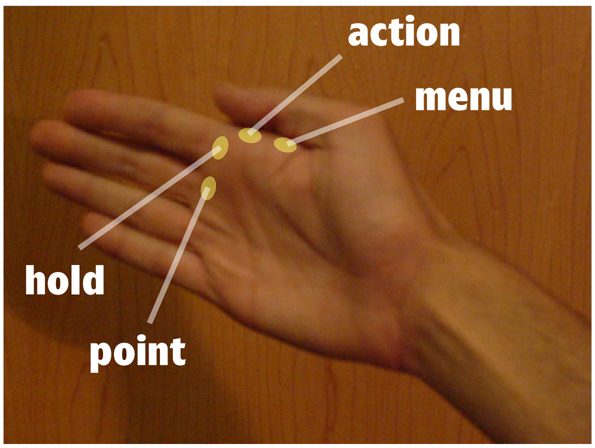
This last observation is a starting point for design. The hand naturally uses the following postures to perform the four functions:



Four postures for the naked hand, representing four classes of action.

If these postures could be natively used to perform all of the functions seen in Part II, it would indeed be a beautiful thing. But there is more work to be done: to properly sense these postures, buttons have to be added to the hand, a slight variation in the grab posture needs to occur, and the grab posture needs to be altered to a less natural form. It is these changes that cause the unmediated interaction naive users dream of to become somewhat more awkward, more arbitrary, and ultimately less direct and culturally supported than the tangible tools presented in Part II.

We have already seen how difficult it can be for continuous sensors to partition input space into state transitions — differentiation between point and hold can be more easily detected by buttons that are depressed when the user is in each posture. The image below shows buttons for each tool. Also shown is a menu button to implement the method of tool remapping presented in Chapter 8.



Hypothetical button positions to support a hands-only interface for two of the four actions classes (point and hold). Two buttons activated by the thumb trigger the active tool and display a halo menu for the active action class.

The user signifies that they are in point mode by depressing the point button. They signify they are in hold mode by depressing either just the hold button, or both the point and hold buttons. In practice, a different digital accessory would be displayed in front of the hand in each of these modes. While this input mapping easily accommodates the addition of the trace posture (where depressing neither the point nor hold button leaves the hand in trace mode, and a tracing accessory appears in front of the hand) it is much more difficult to add the grab posture. While grab mode could be detected using a button on the tip of the index finger, in this posture the user has no way of telling the grab tool to begin its action, or to activate a menu telling it to remap the tool from one mode to the other. Ideally these tasks would be performed by the same buttons that are present in other modes. This is a primary difficulty of trying to map so much functionality onto the hand: there is a fundamental limit to the number of postures that can be clearly differentiated, and moreover some things are very difficult to represent with the hand. A device like Fröhlich's Cubic Mouse is not only difficult to map to the hand, but moreover this mapping threatens to break all the other mappings present in the system.

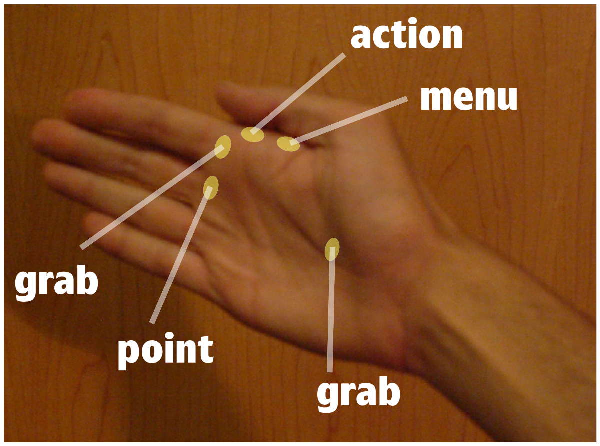
Can we use the pinch to grab, and use the other fingers to trigger an action? Perhaps we could use the very differentiation between a flat-hand and a pinching hand to active a grab?

This differentiation is difficult for users because it is inherently continuous — it is very difficult to place buttons or switches that detect these postures, and as such it is not only difficult to recognize them, but moreover hard for a user to know when they are in which mode (in practice, the user has to play close visual attention to the hand cursor, and is likely to make mode-switching errors in the middle of large gestures where hand posture is difficult to control). Adding a menu posture to this approach is also difficult. Perhaps we sense finger spread?

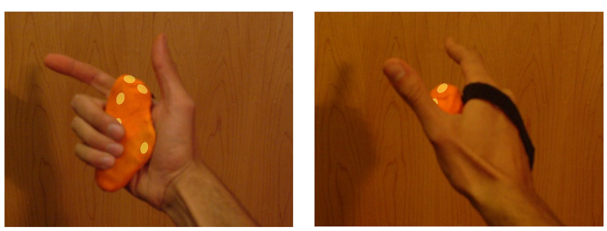
This could be detected by buttons, but note that spreading the fingers in this manner is extremely uncomfortable, especially when the thumb is closed.

It is more reasonable to detect a grab with a button placed in the lower palm, as shown below at left:

We now have a hand sensed by five buttons, three of which determine mode, one action button, and a button to activate a local menu to remap the functionality of each posture.



We now have another difficulty, in that the postures are not very comfortable (the reader is encouraged to run through the motions of pressing the action and menu button in each posture to see this for themselves). In some sense, we are very far from our dream of natural uninhibited action of the hand. This discomfort can be lessened by building an input device that places the buttons so that they require less effort:



While this devices eases input, it takes us even further from the dream of using the hand directly as a tool. Also note that, at this point, the cultural affordance of each posture has been greatly reduced. The act of grabbing in this interface is much less supported than the use of the tongs. The appropriateness of the pointing gesture and its kinesthetic framing are dependent upon a cooperation with the user that is not explicitly enforced.

In this chapter's larger theme of generic and specific interface, the hand interaction is more generic because it uses only one device. This has advantages in that users do not need to manage the location of tools in the work environment. With two copies of this device, all tools are quickly accessed with either hand. This advantage is not without its costs — users need to pay close attention to the position of their hands, in general less attention is needed if physically distinct tools are used. In addition, the hand-button interface is much more arbitrary than the separate tools. Users need to remember what button maps to what function class. Looking at the hand with buttons there is no way for a user to implicitly know which button performs which function. A standard mapping of button to function class is not enforced, application developers are not implicitly encouraged to follow a standard. Most crucially to the larger themes of this work, the single-hand mappings lessen the cultural affordance and kinesthetic framing of input which lends much of the intuition and fluidity to the methods seen in Part II.

10 interface analysis

design choices, spatial mappings, charting the territory

The interactions we have studied form a fuzzy bundle of knowledge. Interfaces are never perfect, always subject to personal preference and cultural situation. The lessons learned from spatial construction overflow into other interfaces, particularly those that too deal with space and tangible input. This chapter contains concepts that these interactions highlight, and illustrates the choices that designers face in constructing coherent interfaces that are well suited to user's intentions.

DIRECT UNION, STRONG AND WEAK DIRECT MANIPULATION

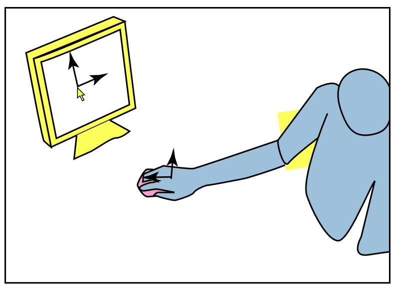
In the example of the raygun, the digital beam of the gun can either match the kinesthetic frame, or not match it. Similarly a mouse's motion can either match the kinesthetic frame (moving the mouse up moves the pointer up, moving it to the right moves the mouse to the right, etc) or not match it. What is the relationship between the frame of the input device, and that of the accessory it is controlling in digital space? The interactions in this text tend to exhibit a **direct union** where the accessory is at a constant position in the tangible tool's local coordinates. When a sensible kinesthetic frame is established, and there is a balance between physical space and digital effector, it is as if the user's body *is* that effector for the duration of the interaction, and the strength of this relationship is at the heart of the immediacy and fluidity users perceived when using the interfaces built upon these interactions.

I introduce this terminology because, while the value of directness has long been known in the interface community [Shneiderman 1983], there is a failure to differentiate between degrees of directness which are quite significant in practice. Direct manipulation refers to the externalization of internal variables so that users can control them explicitly, often through graphic means.

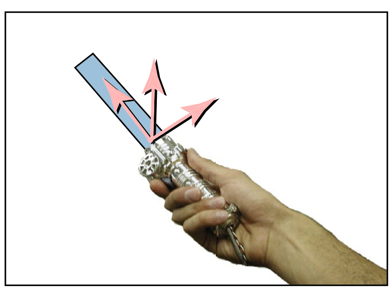
There is also the concept of direct mapping which does not have the strict spatial requirements that direct union describes. Direct mapping refers to the predictability of a relationship between input and action, as seen in a description of a box whose lid controls musical parameters:

> *...an intuitive direct mapping would be to control volume based on the box lid position. (Lid open equals full volume, Lid closed equals no sound). An example of indirect mapping might be choosing a random pattern of notes when the lid position is changed.* [McCarty 2003]

Direct union refers to a tighter link of input and action, in particular an identification of input device space and display space.



In this image, there is a **bijection** between the coordinates of the mouse pointer and the form of the mouse. Note that, in the perception of the user, this map exists only in small localities of interface time — if a user picks up a mouse, and puts it down again, the relationship is reset. This is essential for navigating a large screen with a limited amount of desk space. The WACOM tablet, a tablet which senses a digital pen, makes an explicit differentiation between one-to-one mode, where the bijection is permanent, and relative mode, where only relative translations are applied to the on-screen cursor.



In direct union, an **identity map** exists between the physical input device and the accessory. In this style of interface, a single coordinate system is shared by both physical device and the corresponding element of the display.

The degree of union, seen in the mathematical map between input and display space, partitions direct manipulation into strong and weak forms. Here are some examples along the continuum:

weak direct manipulation

The gasoline and brake pedals of an automobile map to the acceleration and deceleration of the vehicle.

A pen-based input tablet, such as the WACOM tablet, in relative mode. Motions on the tablet are mapped to relative displacements of the mouse cursor. In this interaction, a temporary affine map between pen tip and pointer is built each time the pen contacts the sensing surface.

Pulling a control point in a 3d modeling application such as Maya. In this system, relative 2d displacements are mapped to displacements of the surface along orthogonal axes within the rendered space.

A WACOM tablet in 1-1 mode, where the contact surface's relative coordinates determine the screen-space coordinates of the pointer. This interaction maintains a permanent affine map between display surface and input surface.

Drawing with a remapped accessory, which is related to a phicon by an identity map, although with some amount of spatial separation between phicon and icon.

Drawing with the hand, as in surface drawing. In this interaction the cursor position is identical to that of the hand.

Swinging a baseball bat — very many physical interactions that involve the hand are direct mimicries. In this case, the bat's position is constant with respect to the palm. It is this intimate connection between body and object, often taken for granted, that makes it easy for us to use physical tools as extensions of our own bodies.

strong direct manipulation (direct union)

A way to introduce direct union into the design process is to place input devices first in how an interaction is conceived. In direct manipulation, internal variables are mapped to available input device parameters. Direct union tends to operate in reverse, taking a real-world object and giving it a presence in the digital world.

DESIGN CHOICES

There are a host of issues that go into making effective interactions. A way of understanding the choices involved is to polarize differences in interaction style. The best design is often the resolution of this dialectic: a point somewhere in-between, leading us to a deeper understanding of the choices involved. We have already seen a few such distinctions:

discrete vs. continuous

specific vs. generic

virtual/virtual vs. physical/virtual

strong direct manipulation vs. weak direct manipulation

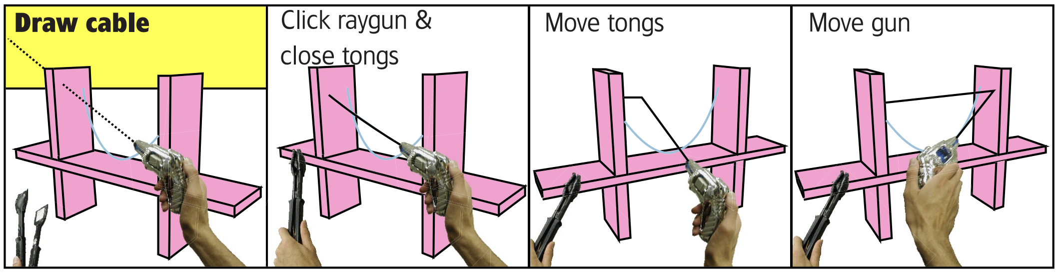
deliberate vs. arbitrary

In all of the design choices that will be discussed here, there are advantages to either end of this spectrum. For example, we have seen the strengths of strong direct manipulation, and we can also note that driving a car with such a direct union (imagine an iconic representation of the vehicle and the street that is controlled inside the vehicle) would afford less control than the current steering wheels and pedals, which are closely coupled to the mechanics of the device. Also note how the indirect style of automobile control allows visual attention to remain focused on the road ahead.

multilocus vs. singlefocus

Having multiple input devices leads naturally to two-handed input. Advantages lie not only in parallelism, but moreover the way that dominant and non-dominant hand combine to perform a task. One of the greatest advantages of two-handed interaction is the ability of the non-dominant hand to establishes a reference frame for the dominant hand's precise input [Hinckley 1997]. As observed by Ullmer and Ishii [Ullmer 1997], the tangible design philosophy naturally supports having multilocus interface, where each tangible controller has an independent locus of control which allows for a natural parallelism in task completion.

In the 3d interfaces for spatial construction, users grab and pull shapes with tongs in the non-dominant hand while polishing with the dominant hand. This not only serves to increase spatial understanding in Hinckley's sense, but moreover allows for dynamic interactions to happen quickly. For example, in a bridge-drawing application (implemented by Yuan Xie at Caltech in 2003), a user can draw cables between two columns without initially having them both in view.



In a singlefocus mouse-based interaction a user has to sequentially rotate the model to see the first connection point, select it, re-rotate the model, and then select the destination point. This is difficult not only because it takes more time than a parallel specification, but moreover because the mouse's existence as bridge-drawing tool is interrupted by a brief period where it's identity is a world-rotating tool. A two-handed interaction with the tongs and raygun fluidly specifies this in parallel, with each tool maintaining a static identity for the duration of the interaction.

Multifocus interaction not only benefits a single user, it also naturally supports multiuser interaction. Multifocus interface naturally supports a vision of ubiquitous augmented reality where everyone on the planet can interact at the same time. I often, when teaching new users how to draw in 3d space, subtly grab and move objects with one set of tongs while they interact with the other input devices. This rarely creates a disruption.

The chief limitation of multilocus interaction is that it does not leave us with a clear linear path of function. This can be seen by first looking at the Undo model presented in Design Patterns [Gamma 1994], then noticing that such commands are not so deterministic in a multilocus world. For applications when a user's actions will be replicated by a single machine (say a user is enacting a tool path, or something), singlefocus interaction more closely mimics the situation at hand, and multilocus interaction is no longer appropriate.

Multilocus interaction, especially in the model of a ubiquitous augmented reality, raises issues of power. As a toy example, consider two users, each with one pair of tongs, each trying to move a box in a different direction. Which user gains control of the box? Any implementation of multilocus interface defines (explicitly or via cultural convention) the resolution of conflicts between loci of control. Networked file systems have long allowed multiple users to act simultaneously on the same data representation. Perhaps the permission-based structure that is so prevalent in this world will be adapted to large-scale immersive spaces? Networked video game spaces tend to solve this by giving each user in the space the ability to act on a limited area of space surrounding their character.

shared-space vs. disjoint

Interactions can either be locally contained, or distributed over a wide area. In early experiments with the hand this issue became quite clear. An early interface triggered the hand's action with the stylus' button. This feels profoundly weird, although it is not clear why. Readers less familiar with 3d interaction can imagine the standard mouse click being replaced by a button in the non-dominant hand. This coupling of button and device is a shared-space interaction, while the splitting of the input between distant points is spatially disjoint.

In some cases disjoint input is natural. For example holding the Ctrl key in the Windows interface allows users to select multiple items (without the Ctrl key, clicking another item will deselect any previously selected item). Is this difference inherent? or learned? or is there another explanation? Perhaps this is because the Ctrl key is not unified in time with the act of selection (it can be held for some period before and after the mouse click). From this second example, we see that not only can disjoint interaction be useful, but that this design choice can lie in time as well as space. Perhaps because the keypress is far away from the mouse click in time, it can also be so in space?

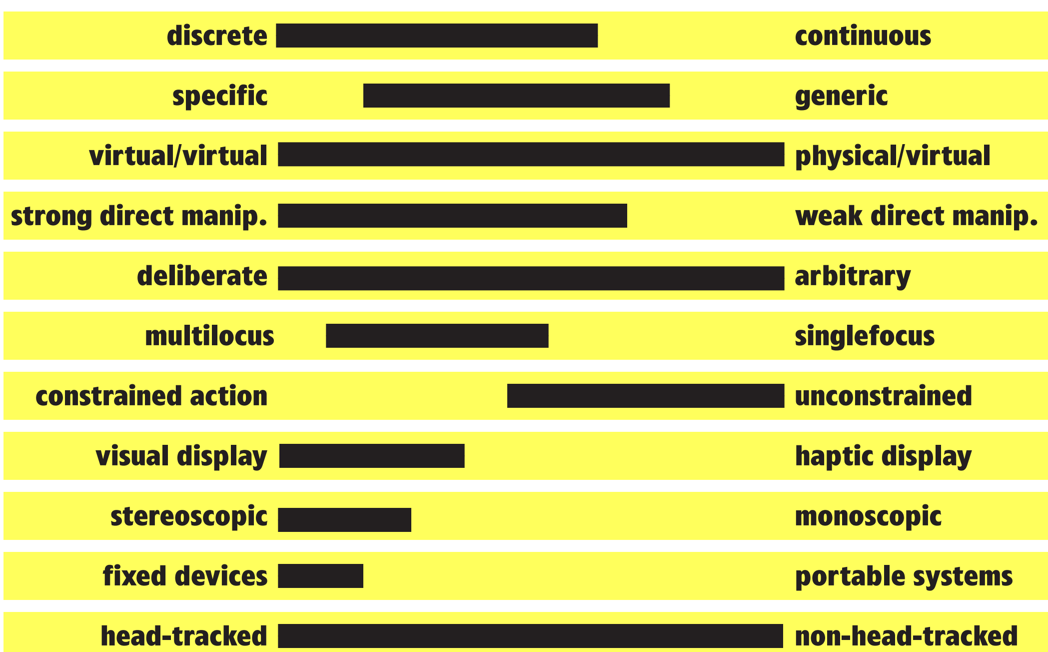
Some interactions naturally benefit from a spatial distribution. The two-phicon Scale action first implemented in 2d by the Bricks interface [Fitzmaurice 1995] and mimicked in this text in 3d with the tongs takes as its basis spatially distributed input: two independent coordinate frames. Thus disjoint interaction is natural for some settings. In some animation settings, temporally distant input can be used to place events on a timeline.

Another aspect of interaction locality lies in the travel required to affect a certain change. We saw this in Chapter 8 when we compared local menus to floating menus and docked menus. This also comes into play when we differentiate the raygun, which selects points at a distance, to the tongs, which select by coexisting with an item. Note that the tongs themselves can have their spatial neighborhood enlarged slightly, this is eased with discretizing visual cues as seen in Chapter 5. The relationship between components of the physical act of input, the phrasing of time within an application, and the relationship between different components that constitute an action greatly affect the fluidity of an interface. In a certain sense, proper treatment of locality defines a notion of fluid interface where juxtaposing greatly varied input, display, and action occurs with ease.

LEVERAGING DESIGN CHOICES TO FURTHER INTERFACE DESIGN

Enumerating design choices lends understanding, both giving consciousness to implicit design choices, and providing a framework with which to analyze existing interface. These concepts can also be used to push design in new directions. Two visual forms are useful in this process: river diagrams, and interface playing cards.

A river diagram is a way to understand the multidimensional interface design space, in particular this diagrammatic form illustrates what has been seen, and what territory remains to be covered. The inspiration for this form is an image constructed by Fitzmaurice, Ishii, and Buxton to demonstrate the portions of the design space analyzed by the Bricks interface [Fitzmaurice 1995]. Here we repurpose this diagram with the choices we have mentioned above, showing in black the river of territory that has been studied in Part II.



The river diagram gives an idea of where we have yet to explore, it shows the whole space. A method that shows certain points in the design space are interaction playing cards. This tool can be made by writing the name of each competing property on one side of a card. To play this game, pick a few cards at random, throw them into the air, and pick up the cards.

Interaction playing cards

A method of randomly sampling the space of design choices.

The exposed faces form a hand. A design experiment is to imagine an interface that has all the properties shown on your cards. Some interesting combinations can arise from point-sampling the multidimensional design space in this fashion. One example is the combination: local interaction / haptic display / unconstrained motion. By simply arranging these possibilities we have to wonder — is all haptic display constrained? We consider such hypothetical interface questions in the next chapter. These methods represent two ways of exploring the multidimensional design space: enumeration and random sampling.

11 speculative interface

what might exist 100 years from now, with an emphasis on spatial construction

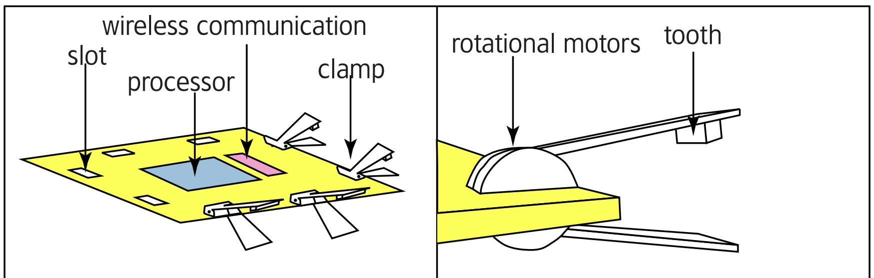
How will humans interact with computers in 100 years? This is impossible to predict, but the act of speculation provides some ideas and motivations. As fuzzy, unpredictable, and dependent on factors out of our control as the future may be, in another sense it is we who create it, and we have some say in how things go. Moreover, a solid notion of where we are heading in the long term allows us to make more effective decisions in the short term.

In this chapter, we play a ridiculous game of speculation. A ridiculous game for a highly interconnected society of exponential population growth. Ridiculous because of technological factors such as Moore's law. Ridiculous because it relies, from time to time, on technology that has yet to be developed, the science of which is not clear. And ridiculous because it has no way of incorporating those surprises, those great discontinuities which history is so fond of throwing in the paths of the unsuspecting.

The conversation is focused on spatial construction, but not limited to this area, and is conducted by first describing hypothetical technologies, and then talking about what they might be used for. While plans for implementing the new devices are sketched out, this chapter is less an argument that certain technologies will be developed, and more of an interface designer's wish list. If we could have any new technologies, what would provide for the most interesting new interactions, the richest new language of spatial form, and the tightest merger between human and machine?

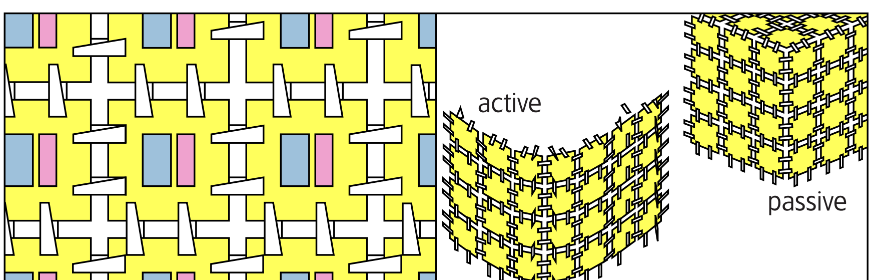
TECHNOLOGY 1: ACTIVE MATERIAL

Perhaps the most rapid arena of change in the past century has been founded on atomic units which have become heavily mass produced. Transistors form logic gates, which form processors, the computers and the networks that they enable. NAND gates alone can be combined to perform any digital operation, and thus designers can forget about the internal properties of the NAND gate and proceed to specify higher-level functions. One of the benefits of atomic units is that they are so friendly to methods of mass production, they can be made so cheaply that they are used for situations even when a custom solution would use fewer total material resources. It is this type of leverage that an atomic material component could bring to virtually every thing humans interact with. What would happen if such an atomic unit was found, not for computation, or for information (note that magnetic storage uses a similar principle), but for material? Such a device might look like this:



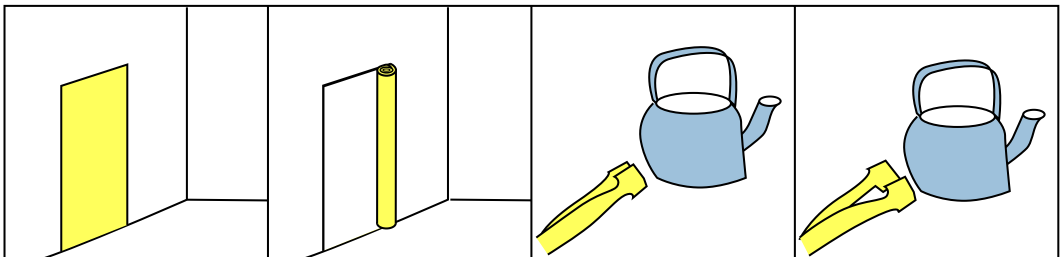
An atomic unit for active material (left), with a side view of one of the active clamps, whose rotational motors are controlled by the processor, which coordinates activities with other atomic units over a wireless network.

This small piece of material has four clamps, which connect to slots on the opposite side of the atomic unit. Motors at the base of the clamps rotate them to allow the angle between two units to flex. A processor at the center of the tile communicates wirelessly, either with the neighboring elements or with a central control device, to change the angles of the clamps and thus change the external appearance of the shape. With multiple copies of this atomic unit, rigid, cubic objects containing right angles between tiles, and flexible objects that curve in one dimension can be constructed.



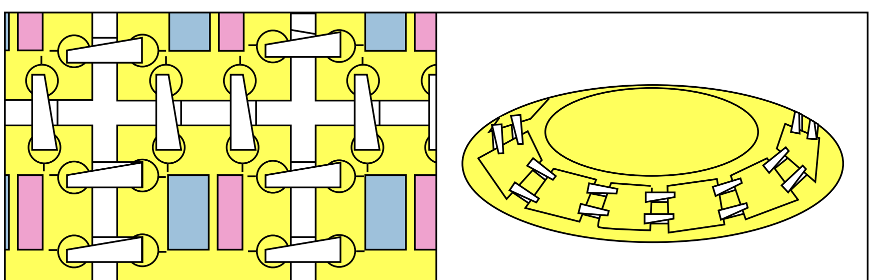
The atomic units link together to tile space. The types of active objects this structure can form is somewhat limited — a flexible sheet with curvature only in one direction is a possibility. Rigid cubic objects can also be made.

While this design can be iterated upon to add further features, we can already see some of the benefits of active material. The first area is in structural engineering — as Berlin showed, active control can be used to increase the structural strength of material. The second area is new aesthetic possibilities — for example, a door could roll up into a tube when it is opened, or elements could customize themselves . Thirdly, this material offers novel solutions to the haptic problem via new input devices and physical display.



With active control of one direction of curvature, the active material could used to build a door that rolls itself up (left), or tongs that change their shape to actively grab a digital object, such as this teapot.

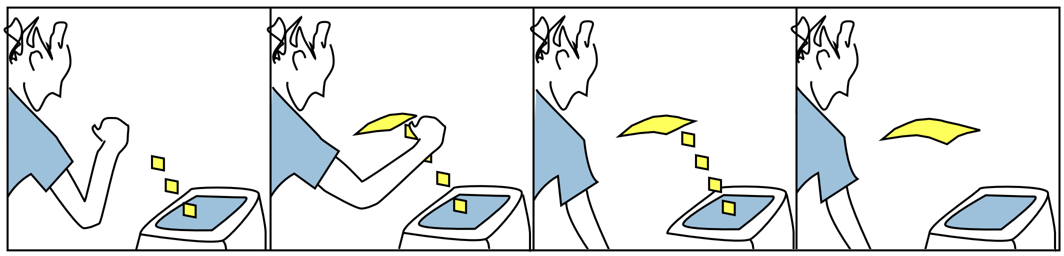
Another iteration on this design gives us a much more flexible unit that is less subject to physical laws, and also provides a particularly elegant solution to the haptics problem. The first change in this iteration is to rework the slots into which the clamps' teeth fit. If the slots are circular (and the teeth are correspondingly round), and moreover they are placed so that they can slide on the surface of the atomic unit, we can represent objects with a much greater range of curvatures.



Making the teeth round, and allowing the holes they fit into to slide over the surface, allows curved structures to be made.

This design is feasible — we note at this point the design just has one type of rotational motor, and one type of linear motor, the only limitation is scale. Another change is somewhat less likely. If controlled fans (imagine miniature helicopters) could be placed on a lightweight atomic unit, it could float in space, providing a distinctly different type of interaction.

For example, surface drawing could be supported with a set of these tiles which float out of a specialized emitter.



Instead of the user seeing a digital representation, they construct a physical representation. This could perhaps be simplified by having the tiles emit from the hand itself. These tiles could run, like ants, in a stream up the users arm and out of the hand.

Note that this requires, not just advanced control, but also an advanced ability for the tiles to sense their environment. This interface not only allows the form to be felt after it is constructed, it can be deformed by directly applying pressure with the whole hand, any other part of the body, or any other tool. This system might be much more portable than projector-based immersive displays. Moreover the resulting forms can be edited after they are removed from the work area. Another set of tiles can duplicate the structure automatically.

This material allows data to be specialized without requiring users to buy specific devices. While, to an extent, we can program a purely visual representation to kinesthetically accommodate the body, this type of material extends the ability to program kinesthetic affordance. For example, a surface/hand could have a haptic and tactile richness that is difficult to foresee with current haptic technologies and current inert material alike. This resistance could be used to facilitate input in much the way the tools described in Part II function. One collection of active material could dynamically change between the four input modes, forming new input devices as a user interacts.

This type of material exhibits a new level of directness that merges a clay-like materiality with dynamic functional specification. Imagine a clay form stretched into a tube.

Bending the tube could specify the placement of a hinge. After performing this action, the clay is actively changed to be more flexible in this area. In this manner, active material promises to add further richness and dynamism to spatial construction.

The active material neatly solves the haptics problem. Researchers have traditionally tried to add haptic feedback by placing a robotic skeleton over the user's body to directly control the underlying human skeleton of the user. One of the few haptic interfaces that does not need to be worn is the magnetic levitation ball [Berkelman 1996]. Extending this approach, this active material provides a much better way to render forces for spatial construction applications.

TECHNOLOGY 2: TOTAL CONTROL OF VISUAL SPACE

Another future direction is the movement away from physicality into purely visual space. Virtual reality has long sought to mimic the natural world, although this technology has in practice been applied to dynamic, creative spaces that break physical laws. The second technology envisions the flexible authoring of visual experience, unbounded by the placement of display [Schkolne 2001b]. Consider the combination of:

- video see-through glasses worn by many users, for the sake of this thought experiment imagine as many users as currently use the internet outfitted with video see-through contact lenses,

- trackers telling such users exactly where they are on the earth,

- a wireless internet connection tied to each user,

- a large geographical database, containing both the geometry of the earth, and humans, cars, and other moving items continuously updated via a widespread network of cameras, which lives on the internet,

- and geometric data for visual forms which one or more users view, composited seamlessly onto the natural world via the see-through displays.

The only barrier to such a system existing is cost. All of the necessary technology exists today, albeit at a smaller scale than necessary to produce these effects. Take caution to note that the feasibility of this idea does not mean it *will* happen. Many exquisite plans for large space stations which provide gravity are also theoretically feasible, but have yet to happen due to enormous cost and dubious financial payback for the parties involved.

This type of system would allow essentially any surface to become a display. This type of effect would also be achievable by embedding pixels in the surfaces of everyday objects.

A pool chair with animated stripes

In this hypothetical space, we see not only the possibility to mimic nature – authoring synthetic woods that burn, synthetic coppers that turn green over time, avatars that wander around the space, and such, but also the opportunity to create objects that are un-natural. For example, the leaf pattern of trees could be mapped to a chair that grows:

A chair that changes with the seasons. The chair evolves over time from left to right: original chair with soil, chair grows leaves during the spring, leaves change colors during the autumn, leaves fall off during late autumn, snow falls in winter, and snow melts so that the process can start over.

Display on the surface of one object can also be linked to cameras elsewhere in the environment. This method could be used as ambient communication, or for visual effect.

Behavior can get deeper into the visual properties of a shape, material can do more than just have a presence in the environment. It can also modulate its environment. Matter can have the property of absence, negating previously constructed forms so that they cannot be seen. This special type of matter can be thought of as a lens which modulates light or other forms of material that pass through it.

A lens integrated into a chair back. Four different lenses, from left to right the lens: blurs the scenery behind it, inverts the colors of the scenery, shows a distorted view of the area behind it (in this case centering its view on the nearest tree), and shows a view of a nearby area.

Unlike photographic negative space, these areas are not defined in terms of location amidst positive form. Instead, they are pure negation – existence that can only be perceived through modification of the already-existing. The territory between absence and presence is filled by lenses that modulate attention, making certain objects more visible.

In the natural world, physical laws determine interactions between objects. Nature can be thought of as a massively parallel computer that resolves inter-object interactions. In VR this decision process is subject to human definition. Objects can physically move through one another without disruption. Visually small objects can affect much larger ones, or shapes can be connected across a vacant space.

A lens that modifies the environment around a chair. This type of lens can help ease the sunny day seen on the left by providing enough shade (center) to read comfortably or prevent sunburn. This type of lens blocks incoming light. Another lens (far right) blocks outgoing light – for example allowing someone to sunbathe in privacy.

A shape could exert a force some distance away from its boundary. The effects of an object can also move beyond the application of force in space to the modification of the functionality of another object. This could be changing its color, its pliability, or its melting point.

Three chairs (left) linked together by an invisible bond. Rotating (center) one chair causes (right) the others to move as well. Do the chairs affect the person in the image? Is this third party forced to take part?

For a single spectator, an object can change as it is viewed from different angles. Multiple viewers can see individualized views that do not share a geometric consistency

This multiview chair has a different visual appearance for each of three different viewers.

Shapes themselves can change with time. Imagine a maze that a participant is walking through. This maze can modify itself as it is being navigated. These interactive geometries ask an experiential understanding which needs more than a single viewing experience to form.

This painting of the frontier of spatial construction at the point of behavior does not depend upon the massive hypothetical shared space where digital and natural coexist, it appears to be a limiting factor even in contemporary research. It is my instinct that strictly material 3d interfaces, such as virtual sculpture, surface drawing, and related interfaces are some of the simplest to create, and it is the boundary where the material hits the info-structural that the interface questions get more difficult.

This visual space requires more sophisticated ways to specify relationships and structures, as designers will need to author experience in addition to form.

TECHNOLOGY 3: ALONG THE FRONTIER OF THE BODY

Much of interaction is dependent upon how the body functions, and genetic engineering could have great impact on the physical interface between human and data. To improve control, perhaps we change the physical structure of the hand so that the input device becomes part of the body. Or perhaps more simply, the hand is augmented with buttons which make it become the device.

While this sounds far-fetched, there is actually much need for this kind of approach, especially amongst a population that is suffering from repetitive-stress injuries from the mismatch between the body's engineering and that of input devices such as keyboards and mice. Devices that are always carried, such as mobile phones, might as well be embedded into the hand itself for added convenience. A further reach of bioengineering might be to add precision, or perhaps a rapid ability to function to the hand — for example, the hand and brain could be engineered to type more quickly, so that the speed of typing could exceed that of speaking even into the best of speech recognition engines.

Bioengineering makes us think of the body as plastic, and thus we have opportunity to consider some interesting solutions. Consider the trade-off between generic and specific input devices. In general, specific devices are better, but difficult to manage. Perhaps we engineer people to have extra hands, this allows for not only a greater variety of input devices, but also for a higher degree of multilocus interface to exist within one user's action. Another possibility, perhaps more imminent on the technological landscape, is to add parts of other animals to humans (via genetic splicing). A frog's tongue allows us to quickly grab objects. Perhaps active material levitates with the wings of insects. The point here is that the material future is wide open. How can we anticipate these changes in interface design?

Whether these interface advances are found through changing the body, or directly interfacing with the mind directly, the central lesson of this text still apply: for spatial construction, the spatial frame around an interaction is the crucial element of design. With appropriate management of this frame, and relation to a digital space, forms can be efficiently managed in a harmony between input and action.

12 closing thoughts

This work has cut a broad swath through a forest of ideas, approaches, and techniques for both immersive spatial construction and tangible user interface. This knowledge both illuminates the ground underneath the path, as well as allowing a particular insight into the surrounding trees. Our focus here has been a new approach to interacting with 3d space, in which custom devices establish certain classes of input into which a wide range of actions are mapped. The immediate application is in 3d interface design: both the local tasks explored here, and in larger applications such as fully immersive room-size virtual reality research. While the focus has been on spatial construction, the message goes much further to other application areas. The techniques here could be applied to video games, where the cultural affordances of the input devices along with the direct, frank manner in which they operate could prove engaging. Motion planning, character animation, and other constructions that occur in time as well as space could benefit from these techniques. The integrated input of the devices shown here, which allow translation and rotation to be specified in time at a quite high bandwidth have quite a bit of potential for temporal construction (animation).

Beyond the application area, the theories proposed and experimental designs supporting these ideas serve to guide interface design. There is a call in the interface community, and society at large for computers that are less scary, more human, and less disembodying — people want to feel connected to digital space. Many researchers approach this by trying to build fuzzy representations that use partial knowledge, or track continuous gestures to inform state transitions, operations and actions. Sadly, many of these techniques fall flat as a machine's ability to reason with such partial information is still quite immature. This work shows that a great deal of emotional connection can be had with much simpler techniques that do not rely on machine intelligence. The use of continuous input is important to this end, but I have found that it is only when it is applied to a continuous change in state that it is effective. Otherwise, as the machine discretizes the input into one action or another, a user actually feels more disenfranchised as the computer often makes the wrong interpretations.

Emotional fluency surrounds devices that are recognizable and have a cultural history. Cultural affordance does not depend on artificial intelligence to bring humans closer to machines. It is my belief that tool familiarity goes beyond knowledge of use into deep history. If one's parents, and parents' parents, and parents' parents' parents used a tool, then there is an implicit familiarity which lends a solid foundation to an experience. This type of intimacy need not be thrown away as we move forward technologically, as the tongs, raygun, handle, and hand-based input demonstrate. The approach to cultural affordance demonstrated throughout this thesis shows how it is possible to bring such cultural ties to a computing environment without sacrificing the wide range of flexible operations that computer users depend upon for hyperefficient interaction.

The balance between many task-specific tools and one or two generic operators is difficult, a growing crisis for the interface community as we develop increasingly sophisticated and specialized modes of interface. In this text, the link between atoms and bits occurs in the space of functionality: tool form is linked to function, not data as it has been in so much prior research on tangible user interface. In this way, tools can be tied to the conceptual methods of a group of practitioners, not the specific task that they are working on. This kind of division already occurs amongst groups of users in different software. For example, programmers deal with text buffers, compiles, linkers, and debuggers. It is these functional elements that serve programmers along their task regardless of the project they are facing, or the language they are using. WIMP-based spatial construction applications deal with rotation widgets, translation widgets, and various tools to move points in a scene and group points for certain types of operations. These tool sets appear similar, regardless of whether someone is making characters for entertainment applications, industrial designs, or buildings. 2d designers have a different set of functions — tools to align objects, layers to control composition, marquee tools to select regions, and handles to stretch and resize 2d objects. Each class of user spends much of their time with the same *software* constructs. In the philosophy of this text, each class of user could spend much of their time with the same interface *hardware* constructs, tailored towards their task. A middle ground between specific and generic can be found in interfaces that are tailored towards broad functions instead of specific requirements of a particular problem's embodiment. Even if 3d interfaces never gain widespread use, these lessons apply to tool design at large.

Another large theme of this work is extending the human body into digital space. Aside from the emotional affects of cultural affordance, and presenting the right sense of general tools, there is a requirement for a tool to integrate well with the digital action that it controls. One concern is to match the parameters of the input space with those of the action and display space, which the concepts of direct union and several observations about activating discrete actions (or components of actions) with discrete inputs. Closer to the body, we observe that there are natural coordinate frames presented by the kinesthetics of the body. Space is not impartial: a physical tool frames the body and mind, and some directions have more meaning than others. Understanding that such a kinesthetic frame exists is crucial to using an input device effectively. With creative luck, ingenuity, application of these principles, and the development of the ideas started in this text into larger concepts and grander applications, we will allow digital space to be manipulated with much of the fluidity that we find in the natural setting, but with the additional advantages of being able to control the physics of these spaces. It is this holistic link to a space of such richness that is the higher aim of this work: to free our ability to conceive, engage the space in which we live.

glossary

The focus of this glossary is to define terms as used in this text, not in their larger meaning. For this reason, some words (e.g. affordance) are defined as applied to input devices, not in their original context.

action A transition in a program's state that is triggered by user input. For example, the motion of a cursor on the screen is an *action*, as are rotations and transformations of a geometric model. Note that many methods of input could be used to trigger any given action.

affordance An aspect of an object that guides its use. Introduced by Gibson [Gibson 1979], the term was given new meaning for interface designers by Donald Norman [Norman 1988], who gave the classic example of door handles. If a push-bar on a door runs horizontally across a door, a user does not know on which side to push the handle. If the handle is placed on the side of the door opposite the hinge, it is clear to push the door on the appropriate side. The placement of the handle is an *affordance*, one says the second option *affords* a user pushing the door on the correct side.

arbitrary An interface with a weak correspondence between inputs and actions is said to be *arbitrary*. Arbitrary interfaces have portions that are not clearly related to their function. Perhaps the easiest test to see if an interface has arbitrary interactions is to swap the inputs that result in an action. If no meaning is lost, then those inputs are arbitrarily mapped to their actions.

cultural affordance A guide to using an input device provided by prior use of a similar device. For example, the way to use the tongs (to grab objects) is based on the prior use of tongs in the kitchen, barbecue, by blacksmiths, etc.

deliberate An interface with a strong correspondence between inputs and actions is said to be *deliberate*. The opposite of *arbitrary*.

direct union An identity map between input space and display space. When a *direct union* is implemented, the digital space appears to be an extension of physical space.

discrete action An *action* (defined above) that specifies a discete state-change in an application. Commands such as Save and Quit are *discrete actions*, while most geometric operations are continuous actions. The methods in this text are primarily suited for continuous actions.

frame A local coordinate system for a geometric object.

halo menu A set of options presenting themselves as a ring in space around an input device. *Halo menus* are roughly equivalent to 3d pie menus and 3d marking menus, both of which were originally created in the 2d setting.

kinesthesia The sense that detects bodily position, weight, or movement of the muscles, tendons, and joints. [Dictionary.com 2003]

kinesthetic framing The physical aspects of a hand-held device which affect the position and motion of the body to both suggest certain spatial relationships and guide action. An aspect of affordance, kinesthetic framing has both mental and physical effects on tool use. The term frame has a double-connotation, referring both to the coordinate frames surrounding a tool and the mental frame of the user during tool use.

phicon A combination of physical and icon, a phicon is a physical device that represents a type of actions. Coined by Ullmer [Ullmer 1997]. Largely interchangeable with prop, tangible tool, and tangible controller.

prop A physical device that supports an interaction. Used by Hinckley to describe his early physical interfaces [Hinckley 1994]. Largely interchangeable with phicon, tangible tool, and tangible controller.

proprioception The unconscious perception of movement and spatial orientation arising from stimuli within the body itself. [Dictionary.com 2003]

reframe An *action* which changes the *frame* of an object. Translating and rotating an object in space are reframe actions, as is scaling it.

spatial construction Creation of geometric artifacts. *Spatial construction* refers to a large number of problems in design, art, and engineering whose solution is some object with coordinates in space.

surface drawing A method for creating surfaces in 3d space by using the hand to directly trace shapes. First described in [Schkolne 1999].

tangible interface (TUI) A method for interacting with computers where physical devices are used as inputs and outputs. Coined by Hiroshi Ishii, this style of interface design refers to levels of touch-customization that are greater than that found in mice.

virtual reality A method of human-computer interface where a digital model of space is presented to the user at all times with the intent to immerse the user in that space. Coined by Jaron Lanier in the late 1980's, this term had precursors such as Krueger's concept of Artificial Reality [Krueger 1983]. The goal of *virtual reality* is a method of computer interface that is indistinguishable from non-computational existence.

references

[ANDERSON 2000] David Anderson, James L. Frankel, Joe Marks, Aseem Agarwala, Paul Beardsley, Jessica Hodgins, Darren Leigh, Kathy Ryall, Eddie Sullivan, and Jonathan S. Yedidia, Tangible Interaction + Graphical Interpretation: A New Approach to 3D Modeling, *Proceedings of SIGGRAPH 2000*, 393–402.

[ANGUS 1995] I. G. Angus and H. A. Sowizral, Embedding the 2D Interaction Metaphor in a Real 3D Virtual Environment, *SPIE Proceedings Vol 2409: Proceedings of Stereoscopic Displays and Virtual Reality Systems,* 282–293, 1995.

[APPLE 1992] Apple Computer, *Macintosh Human Interface Guidelines*, Addison-Wesley, Reading, MA, 1992.

[ARNHEIM 1969] Rudolf Arnheim, *Visual Thinking,* University of California Press, 1969.

[BAJURA 1992] Michael Bajura, Henry Fuchs, and Ryutarou Ohbuchi, Merging Virtual Objects with the Real World: Seeing Ultrasound Imagery Within the Patient, *Proceedings of Siggraph 1992*, 203–210.

[BALAKRISHNAN 1999] Ravin Balakrishnan and Ken Hinckley, The Role of Kinesthetic Reference Frames in Two-Handed Input Performance, *Proceedings of UIST 1999*, 171–178.

[BATTER 1972] J. J. Batter and F.P. Brooks, Jr., GROPE-1: A Computer Display to the Sense of Feel, *Information Processing 71*, North-Holland Publishing Co., Amsterdam, 759–763.

[BERKELMAN 1996] P. J. Berkelman, Z. J. Butler, and R. L. Hollis, Design of a Hemispherical Magnetic Levitation Haptic Interface Device, ASME IMECE, Atlanta, November 17–22, 1996, *DSC-Vol. 58*, 483–488.

[BÉZIER 1970] Pierre E. Bézier, *Emploi des machines a commande numerique*, Masson et Cie, 1970.

[BOZDOC 2003] Marian Bozdoc, The History of CAD, *http://mbinfo.mbdesign.net/CAD-History.htm*, 2000–2003

[BRODY 1999] Bill Brody and Chris Hartman, BLUI, a Body Language User Interface for 3d Gestural Drawing, *SPIE Proceedings Vol 3644: Human Vision and Electronic Imaging IV*, 1999.

[BROOKHAVEN 2003] Brookhaven National Laboratory, Video Games – Did they Begin at Brookhaven?, *DOE Research and Development Accomplishments*, http://www.osti.gov/accomplishments/videogame.html, 2003

[CALLAHAN 1988] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman, An Empirical Comparison of Pie vs. Linear Menus, *Proceedings of CHI 1988*, 95–100.

[CHEN 1999] Chaomei Chen, *Information Visualization and Virtual Environments*, Springer Verlag, 1999.

[CHEN 1991] J. Chen and N. C. Seeman, The Synthesis from DNA of a Molecule with the Connectivity of a Cube, *Nature*, 350:631–633, 1991.

[CONNER 1992] D. Brookshier Conner, Scott S. Snibbe, Kenneth P. Herndon, Daniel C. Robbins, Robert C. Zeleznik, and Andries van Dam, Three-Dimensional Widgets, *Proceedings of the 1992 Symposium on Interactive 3D Graphics*, 183–188.

[CRUZ-NEIRA 1992] Carolina Cruz-Neira, Daniel J. Sandin, Thomas A. DeFanti, Robert V. Kenyon, and John C. Hart, The CAVE: Audio Visual Experience Automatic Virtual Environment. *Communications of the ACM*, 35:6, 67–62, June 1992.

[DE CASTELJAU 1959] Paul de Casteljau, *Outillages Méthodes Calcul*, Technical report, Citröen Automobiles, Paris, 1959.

[DEERING 1995] Michael F. Deering, HoloSketch: A Virtual Reality Sketching/Animation Tool, *ACM Transactions on Computer-Human Interaction*, 2:3, 220–238, September 1995.

[DESBRUN 1999] Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H. Barr, *Implicit Fairing of Irregular Meshes Using Diffusion and Curvature Flow*, *Proceedings of SIGGRAPH 1999*, 317–324.

[DICTIONARY.COM 2003] http://www.dictionary.com.

[EHRSSON 2000] H. Henrik Ehrsson, Anders Fagergren, Tomas Jonsson, Göran Westling, Roland S. Johansson, and Hans Forssberg, Cortical Activity in Precision- Versus Power-Grip Tasks: An fMRI Study, *Journal of Neurophysiology*, 83:1, 528–36, 2000.

[FEINER 1993] Steven Feiner, Blaire MacIntyre, Marcus Haupt, and Eliot Solomon, Windows on the World: 2D Windows for 3D Augmented Reality, *Proceedings of UIST 1993*, 145–155.

[FEINER 1997] Steven Feiner, Blaire MacIntyre, Tobais Höllerer, and Anthony Webster, A Touring Machine: Prototyping 3D Mobile Augmented Reality Systems for Exploring the Urban Environment, *Proceedings of the International Symposium on Wearable Computers 1997*, 74–81.

[FITZMAURICE 1995] George W. Fitzmaurice, Hiroshi Ishii, and William Buxton, Bricks: Laying the Foundations for Graspable User Interfaces, *Proceedings of CHI 1995*, 442–449.

[FITZMAURICE 1996] George W. Fitzmaurice, *Graspable User Interfaces*, PhD Thesis, University of Toronto, 1996.

[FLASAR 2000] Jan Flasar, 3d Interaction in Virtual Environments, *Central European Seminar on Computer Graphics for Students*, 2000.

[FRAZER 1980] J. H. Frazer, J. M. Frazer, and P. A. Frazer. Intelligent Physical Three-Dimensional Modelling System, *Proceedings of Computer Graphics 80*, 359—370.

[FRAZER 1994] J. H. Frazer, *An Evolutionary Architecture*, Architectural Association, London, 1994.

[FRÖHLICH 2000] Bernd Fröhlich and John Plate, The Cubic Mouse: A New Device for Three-Dimensional Input. *Proceedings of CHI 2000*, 526–521.

[GALYEAN 1991] Tinsley A. Galyean and John F. Hughes. Sculpting: An Interactive Volumetric Modeling Technique, *Proceedings of SIGGRAPH1991* , 267-274.

[GAMMA 1994] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, Reading, MA, USA, 1994.

[GIBSON 1979] James J. Gibson, *The Ecological Approach to Visual Perception*, Houghton Mifflin, Boston, 1979.

[GORBET 1998] Matthew G. Gorbet, Maggie Orth, and Hiroshi Ishii. Triangles: Tangible Interface for Manipulation and Exploration of Digital Information Topography, *Proceedings of CHI1998*, 49−56.

[GUIARD 1987] Yves Guiard, Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model, *Journal of Motor Behavior*, 19:4, 486-517, 1987.

[HINCKLEY 1994] Ken Hinckley, Randy Pausch, John C Goble, and Neal F Kassell, Passive Real-World Interface Props for Neurosurgical Visualization. *Proceedings of CHI1994*, 452-458.

[HINCKLEY 1997] Ken Hinckley, *Haptic Issues for Virtual Manipulation*, PhD Thesis, University of Virginia, 1997.

[IGARASHI 1999] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka, Teddy: A Sketching Interface for 3D Freeform Design, *Proceedings of SIGGRAPH1999*, 409-416.

[IGARASHI 2002] Takeo Igarashi and John F. Hughes, Clothing Manipulation, *Proceedings of UIST 2002*, 91-100.

[JACOB 1994] Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and M. Preston Mullen, Jr., Integrality and Separability of Input Devices, *ACM Transactions on Computer-Human Interaction*, 1:1, 3-26, 1994.

[KAWAI 1996] Yoshihiro Kawai and Fumiaki Tomita, Interactive Tactile Display System: A Support System for the Visually Disabled to Recognize 3D Objects, *Proceedings of the Second Annual ACM Conference on Assistive Technologies*, 45-50, 1996.

[KEEFE 2001] Dan F. Keefe, Daniel Acevedo Feliz, Tomer Moscovich, David H. Laidlaw, and Joseph J. LaViola, Jr., CavePainting: A Fully Immersive 3D Artistic Medium and Interactive Experience, *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, 85-93.

[KIM 2000] Namgyu Kim, Gerard Jounghyun Kim, Chan-Mo Park, Inseok Lee, and Sung H. Lim, Multimodal Menu Presentation and Selection in Immersive Virtual Environments, *IEEE Virtual Reality 2000*, 281-288.

[KRUEGER 1983] Myron Krueger, Artificial Reality, Addison-Wesley, 1983.

[KRUEGER 1985] Myron Krueger, VIDEOPLACE - An Artificial reality, *Proceedings of CHI1985*, 35-40.

[KRUEGER 1990] Myron Krueger, M. W., Artificial Reality II, Addison-Wesley, 1990.

[KRUEGER 2000] Myron Krueger, Personal Communication

[KRÜGER 1994] Wolfgang Krüger and Bernd Fröhlich. The Responsive Workbench, *IEEE Computer Graphics and Applications*, 12-15, May 1994.

[KURTENBACH 1993] Gordon Kurtenbach, and William Buxton, The Limits of Expert Performance Using Hierarchical Marking Menus, *Proceedings of CHI1993*, 35-42.

[KURTENBACH 1994] Gordon Kurtenbach, and William Buxton, User Learning and Performance with Marking Menus, *Proceedings of CHI1994*, 258-264.

[LLAMAS 2003] Ignacio Llamas, Byungmoon Kim, Joshua Gargus, Jarek Rossignac and Chris D. Shaw, Twister: A Space-Warp Operator for the Two-Handed Editing of 3D Shapes, *Proceedings of Siggraph 2003*.

[LIANG 1993] Jiandong Liang and Mark Green, JDCAD: A Highly Interactive 3D Modeling System, *3rd International Conference on CAD and Computer Graphics*, Beijing, China, 217-222, August 1993.

[MCCARTY 2003] John McCarty, HCI Projects from Music 250a & 250b, http://ccrma-www.stanford.edu/~jmccarty/hcipage.htm, Stanford University 2003.

[MCDONNELL 2001] Kevin T. McDonnel, Hong Qin, and Robert A. Wlodarczyk, Virtual Clay: A Real-Time Sculpting System with Haptic Toolkits, *2001 Symposium on Interactive 3D Graphics*, 179-190.

[MCKIM 1980] Robert H. McKim, *Thinking Visually*, Dale Seymour Publications, 1980.

[MEYER 2003] Mark Meyer, personal communication, July 2003.

[MILLER 1956] George A. Miller, The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information, *Psychological Review*, 63, 81-97, 1956.

[MINE 1997] Mark Mine, Frederick Brooks, Carlo Sequin, Moving Objects in Space: Exploiting Proprioception In Virtual-Environment Interaction, *Proceedings of SIGGRAPH1997*.

[NAPIER 1956] J.R. Napier, The Prehensile Movements of the Human Hand, *Journal of Bone Joint Surgery*, 38B, 902–913, 1956.

[NEWTON-DUNN 2002] Henry Newton-Dunn, Hiroaki Nakano, and James Gibson, *Block Jam*, shown at Emerging Technologies, SIGGRAPH 2002.

[NORMAN 1988] Donald A. Norman, *The Psychology of Everyday Things*, Basic books, 1988.

[PIERCE 1997] Jeffrey S. Pierce, Andrew Forsberg, Matthew Conway, Seung Hong, and Robert Zeleznik, Image Plane Interaction Techniques in 3D Immersive Environments., *1997 Symposium on Interactive 3D Graphics*, 39-43.

[PIERCE 1999] Jeffrey S. Pierce, Brian C. Stearns, amd Randy Pausch, Voodoo Dolls: Seamless Interaction at Multiple Scales in Virtual Environments, *1999 Symposium on Interactive 3D Graphics*, 141-145.

[POUPYREV 1997] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR, *Proceedings of UIST1996*, 79-80, 1996.

[RAZZAQUE 2001] Sharif Razzaque, Zachariah Kohn, and Mary C. Whitton, Redirected Walking, Department of Computer Science Technical Report TR01-007, University of North Carolina, 2001.

[SACHS 1991] Emanuel Sachs, Andrew Roberts, David Stoops, 3-Draw: A Tool for Designing 3D Shapes, *IEEE Computer Graphics & Applications*, Nov. 1991, 18-26.

[SCHKOLNE 1999] Steven Schkolne, *Surface Drawing: The Perceptual Construction of Aesthetic Form*, M. Thesis, Caltech, 1999.

[SCHKOLNE 2001] Steven Schkolne, Michael Pruett, and Peter Schröder, Surface Drawing: Creating Organic 3d Shapes with the Hand and Tangible Tools, *Proceedings of CHI 2001* , 261-268.

[SCHKOLNE 2001B] Steven Schkolne, and Dan F. Keefe, Visual Space After Virtual Reality, *Proceedings of YoungVR 2001,* KAIST, Daejeon, Korea.

[SCHKOLNE 2002] Steven Schkolne, Drawing with the Hand in Free Space, *Leonardo,* 35:4, August 2002.

[SHNEIDERMAN 1983] Ben Shneiderman, Direct Manipulation: A Step Beyond Programming Languages, *IEEE Computer*, 16:8, 57–69, August 1983.

[SEDERBERG 1986] T. W. Sederberg and S. R Parry, Free-Form Deformation of Solid Geometric Models, *Proceedings of SIGGRAPH 1986*, 151-160.

[SENSABLE 2003] SensAble Technologies, personal communication, 2003.

[SHAW 1994] Chris Shaw and Mark Green, Two-Handed Polygonal Surface Design, *Proceedings of UIST 1994*, 205-212.

[SHOEMAKE 1994] Ken Shoemake, Arcball Rotation Control, *Graphics Gems IV*, 175-192, Academic Press, 1994.

[SHOUP 1975] Richard G. Shoup, SUPERPAINT Program, Xerox Palo Alto Research Center memo, Feb 5, 1975.

[SINDEN 1994] Richard R. Sinden, *DNA Structure and Function*, Academic Press, 1994.

[SLATER 1999] Martin Usoh, Kevin Arthur, Mary C. Whitton, Rui Bastos, Anthony Steed, Mel Slater, and Frederick P. Brooks Jr., Walking > Walking-in-Place > Flying, in Virtual Environments, *Proceedings of SIGGRAPH 1999*, 359–364.

[SMITH 1997] Alvy Ray Smith, Digital Paint Systems: Historical Overview, Microsoft Technical Memo 14, May 30, 1997.

[STELARC] Stelarc, performance art, sculpture, and interactive media 1968-2003, http://www.stelarc.va.com.au/.

[STOAKLEY 1995] Richard Stoakley, Matthew Conway, Randy Pausch, Virtual Reality on a WIM: Interactive Worlds in Miniature, *Proceedings of CHI 1995*, 265-272.

[SUTHERLAND 1963] Ivan Sutherland, SKETCHPAD, A Man-Machine Graphical Communication System, *Proceedings of the Spring Joint Computer Conference*, 329-346, 1963.

[SUTHERLAND 1968] Ivan Sutherland, A Head-Mounted Three-Dimensional Display, *1968 Fall Joint Computer Conference, AFIPS Conference Proceedings*, 33, 757-764.

[TAN 2001] Desney S. Tan, George G. Robertson, and Mary Czerwinski, Exploring 3D Navigation: Combining Speed-Coupled Flying with Orbiting, *Proceedings of CHI 2001* , 498-505.

[TAUBIN 1995] Gabriel Taubin, A Signal Processing Approach to Fair Surface Design, *Proceedings of SIGGRAPH 1995*, 351-358.

[TAYLOR 1993] Russell M. Taylor, Warren Robinett, Vernon L. Chi, Frederick P. Brooks, Jr., William V. Wright, R. Stanley Williams, and Erik J. Snyder, The Nanomanipulator: A Virtual-Reality Interface for a Scanning Tunneling Microscope, *Proceedings of SIGGRAPH 1993,* 127-134.

[ULLMER 1997] Brygg Ullmer, and Hiroshi Ishii, The metaDESK: Models and Prototypes for Tangible User Interfaces, *Proceedings of UIST 1997*, 223-232.

[ULLMER 1998] Brygg Ullmer, Dylan Glas, and Hiroshi Ishii, mediaBlocks: Physical Containers, Transports, and Controls for Online Media, *Proceedings of SIGGRAPH 1998*, 379-386.

[UNDERKOFFLER 1999] John Underkoffler, Brygg Ullmer, and Hiroshi Ishii, Emancipated Pixels: Real-World Graphics in the Luminous Room, *Proceedings of SIGGRAPH 1999*, 385-392.

[WANG 1995] Sidney Wang and Arie Kaufman, Volume Sculpting, *1995 Symposium on Interactive 3D Graphics*, 151-156.

[WESCHE 2001] Gerold Wesche and Hans-Peter Seidel, FreeDrawer: A Free-Form Sketching System on the Responsive Workbench, *Proceedings of the 2001 ACM Symposium on Virtual Reality Software and Technology*, 167-174.

[WESCHE 2003] Gerold Wesche, The ToolFinger: Supporting Complex Direct Manipulation in Virtual Environments, *Proceedings of the ACM/Eurographics Workshop on Virtual environments 2003*, 39-45.

[WINFREE 1998] Erik Winfree, *Algorithmic Self-Assembly of DNA*, PhD Thesis, Caltech, 1998.

[WINFREE 1998B] Erik Winfree, Furong Liu, Lisa Wenzler, Nadrian C. Seeman, Design and Self-Assembly of Two-Dimensional DNA Crystals, *Nature* 394, 539-544, Aug. 6, 1998.

[WINTER 2003] David Winter, *PONG-Story: Magnavox Odyssey—the first video game*, http://www.pong-story.com/odyssey.htm, 1996-2003.

[YOU 1999] Suya You, Ulrich Neumann, and Ronald Azuma., Hybrid Inertial and Vision Tracking for Augmented Reality Registration, *Proceedings of IEEE Virtual Reality 1999*, 260-267.

[ZELEZNIK 1996] Robert C. Zeleznik, Kenneth P. Herndon, and John F. Hughes, SKETCH: An Interface for Sketching 3D Scenes, *Proceedings of SIGGRAPH 1996*, 163–170.

[ZELEZNIK 2001] Robert C. Zeleznik, Timothy Miller, and Andrew Forsber, Pop Through Mouse Button, *Proceedings of UIST 2001* .

[ZELEZNIK 2002] Robert C. Zeleznik, Joseph J. LaViola, Jr., Daniel Acevedo, and Dan F. Keefe, Pop Through Buttons for Virtual Environment Navigation and Interaction, *Proceedings of Virtual Reality 2002*.