Augmenting Visual Feedback Using Sensory Substitution

by

Eugene Greene

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2011

Abstract

Direct interaction in virtual environments can be realized using relatively simple hardware, such as standard webcams and monitors. The result is a large gap between the stimuli existing in real-world interactions and those provided in the virtual environment. This leads to reduced efficiency and effectiveness when performing tasks. Conceivably these missing stimuli might be supplied through a visual modality, using sensory substitution. This work suggests a display technique that attempts to usefully and non-detrimentally employ sensory substitution to display proximity, tactile, and force information.

We solve three problems with existing feedback mechanisms. Attempting to add information to existing visuals, we need to balance:

- not occluding the existing visual output;
- not causing the user to look away from the existing visual output, or otherwise distracting the user; and
- displaying as much new information as possible.

We assume the user interacts with a virtual environment consisting of a manually controlled probe and a set of surfaces.

Our solution is a pseudo-shadow: a shadow-like projection of the user's probe onto the surface being explored or manipulated. Instead of drawing the probe, we only draw the pseudo-shadow, and use it as a canvas on which to add other information. Static information is displayed by varying the parameters of a procedural texture rendered in the pseudo-shadow. The probe velocity and probe-surface distance modify this texture to convey dynamic information. Much of the computation occurs on the GPU, so the pseudo-shadow renders quickly enough for real-time interaction.

As a result, this work contains three contributions:

- a simple collision detection and handling mechanism that can generalize to distance-based force fields;
- a way to display content during probe-surface interaction that reduces occlusion and spatial distraction; and
- a way to visually convey small-scale tactile texture.

Acknowledgements

Thank you to my family, to my supervisor, to my friends, to my labmates, to my readers, to my housemates, to my teachers, and to NSERC. Thank you for being interesting, for being funny, for being supportive, for being wise, for being weird, for being encouraging, for being kind, for being good cooks, for being good people, and for all the money.

Dedication

This thesis is dedicated to anyone actually reading it. I hope you find what you're looking for (even if it's ideas for a dedication page).

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Many aspects of real-world stimuli are missing in simple virtual reality environments, which lack specialized hardware like stereo output or haptic devices. For example, our eyes receive no information about convergence, accommodation, or retinal disparity, three cues important for stereo vision. Our skin receives no information about pressure, vibration, temperature, or force. People rely on these cues in the real world without necessarily noticing them, and our actions become less effective and efficient when they are missing in a virtual environment. Xin, Burns, and Zelek [92, page 27] discuss the lack of visual and haptic feedback in the context of laparoscopic surgery:

> The indirect visual feedback from laparoscopic surgery causes reduced depth perception and poor hand-eye coordination and does not provide enough force feedback information about the state of tissue unless it has been cut or torn. The reduced haptic feedback gives surgeons less natural and direct information about their applied force. Surgeons are forced to infer the state of operable tissue through cues such as tissue depression and the presence of blood.

Hu et al. [31] describe an experiment in which this is again evident. Subjects were to place a virtual cylinder on a virtual table while holding a real cylinder as an input device (moving the real cylinder correspondingly moved the virtual cylinder). A head-mounted display provided feedback while toggling the presence of stereo vision, shadows, and interreflections. Subjects tended to be more accurate and precise while receiving stereo output, with shadows and interreflections further helping the subjects.

Arsenault and Ware [5] provide another example. In this experiment, all visual output was in stereo. They asked subjects to perform a tapping task, alternately tapping the tops of two virtual cylinders. Measuring task completion times, they found that force feedback and head-tracked perspective projection both improved task speed. Force feedback also resulted in fewer errors (failing to hit a cylinder).

Is it possible to convey this missing information through other modalities, using only hardware that is customarily available? Sensory substitution is the transformation of inputs from one modality (pressure, for example) into those suitable for another [7] (vision, for example). This thesis thus discusses how to convert this kind of missing information to information suitable for vision, in a useful and non-detrimental manner.

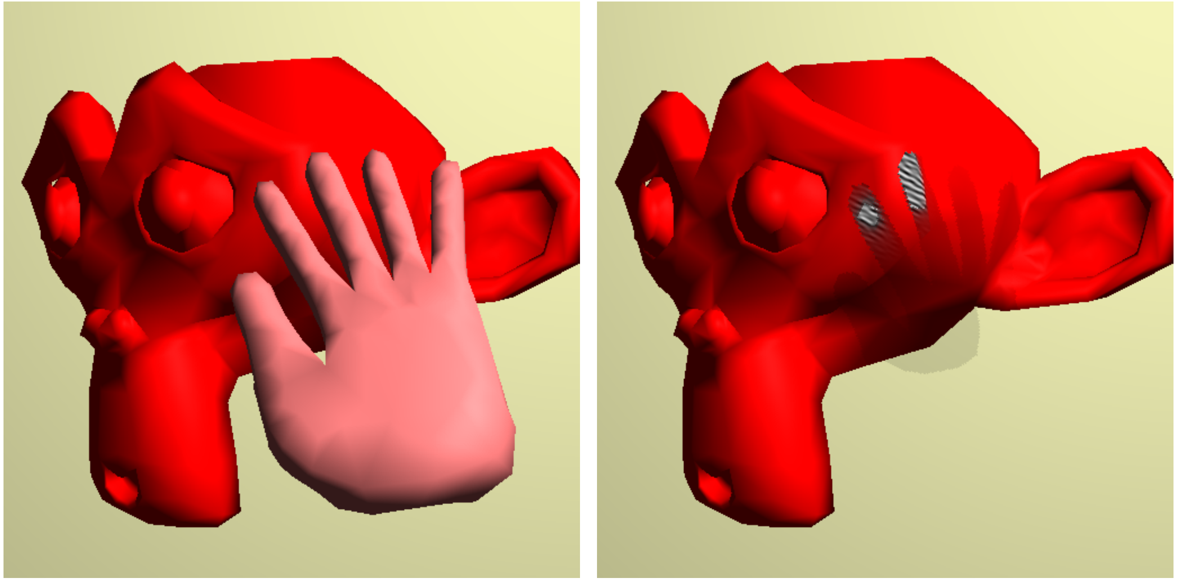
The context for this work is manual interaction in 3D virtual environments. The environment consists of a set of virtual *surfaces*, each with a set of properties that correspond to their real-world counterparts. The user interacts with the virtual surfaces through a virtual *probe*. Figure 1.1a shows a hand-like probe interacting with a monkey-head surface. We assume that the user controls the probe using a manual input device.

So far we have shown the importance of stereo vision and haptic feedback. Chapter 2 lists some of the information that current work attempts to visually display; this information turns out to be related to stereo vision and haptic feedback. Many researchers augment visual output with proximity or distance information; in the real world, stereo vision plays an important part in delivering this. Proprioception is the sense of the relative positions of body parts, and haptic feedback helps deliver this information in the real world. Researchers use *pseudo-haptics* to visually convey this information. Pseudo-haptics fools users into believing visual information about probe position, even when it conflicts with the natural proprioceptive information they receive from their bodies. Haptic feedback also delivers tactile texture information, but researchers have not attempted to visually display small-scale tactile texture (the elements of which are smaller than $0.1mm$).

We thus focus on augmenting output with information about

- proximity and contact,
- tactile texture properties, and
- proprioceptive force.

Figure 1.1b shows our solution: a *pseudo-shadow*. Instead of drawing the probe, it draws a shadow-like projection of the probe on the surface. The intensity of the shadow denotes proximity and contact: the more opaque the shadow is, the closer the probe is. We illustrate tactile texture with a procedural visual texture. We also implement pseudo-haptic feedback to visually convey proprioceptive force information.



(a) No extra feedback

(b) A pseudo-shadow

Figure 1.1: A hand-like probe interacting with a monkey-head surface, with and without a pseudo-shadow

This work makes three contributions, which arise as a result of implementing the pseudo-shadow. We describe a simple collision detection and handling mechanism; we show how to non-detrimentally add visual content during probe-surface interaction; and we introduce a way to visually convey small-scale tactile texture. Before discussing the solution, the following sections suggest why it makes sense to provide this kind of information through vision.

1.1 Sensory Substitution

Bach-y-Rita [7] introduced the idea of sensory substitution in the context of displaying visual information through touch. The Tactile Vision Sensory Substitution (TVSS) system provides electrical or vibratory stimulation through an array of cells in contact with the skin of the tongue, back, abdomen, or thigh. The TVSS system was designed to aid those with visual impairment, but researchers have implemented systems to substitute for other modalities. In particular, this work seeks to do the opposite: displaying tactile information through vision (see Figure 1.2).

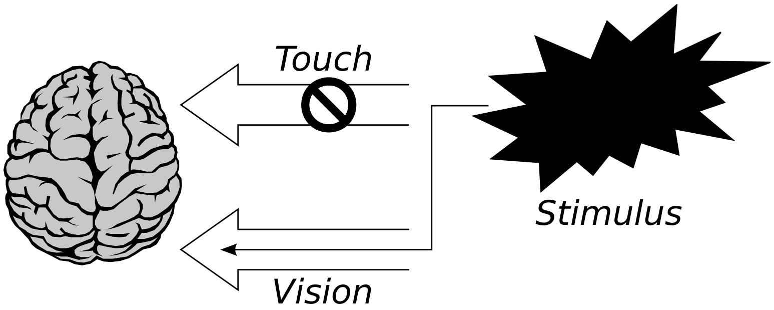


Figure 1.2: Substituting visual information to display tactile information

In general, sensory substitution is the transformation of inputs from one modality into those suitable for another. Examples of substituting for audition occur in Reed, Durlach, and Bradia's [72] review. In particular, they discuss substituting for audition using tactile feedback. For an example of substituting for force feedback, see Riso's [76] work on sub-cutaneous electrical stimulation to provide feedback from limb prostheses. It is even possible to go beyond the natural senses. For example, Nagel et al. [65] built a belt that provides a magnetic sense, informing the wearer of the direction of north.

The TVSS system has been proven to work, and to be useful. In fact, mechanisms such as braille and walking sticks provide visual information through touch. The problem with displaying vision using touch, though, is that visual information needs to be "reduced" to something suitable for touch. Bach-y-Rita discusses how, "This highly complex 'visual' world can thus be reduced, by selective processes, to manageable proportions, allowing the input to be mediated by the somesthetic system," [8, page 33] and how, "The central processing mechanisms may be able to extract a greater percentage of the available information from the lower resolution tactile display than is normally expected from the higher resolution visual display" [8, page 127]. These challenges associated with the TVSS system indicate that it might, in fact, be easier to display touch using vision, than the other way around.

1.2 Vision and Touch

Vision and touch are not completely different. They share properties and, perhaps, mechanisms that appear to facilitate information transfer between them. Touch and vision have similar neurological pathways, physiological responses, and internal representations of form, motion, and texture [30].

There are two neurological pathways with similar properties in vision and touch. Both modalities have *sustained pathways*, with high spatial resolutions and low temporal resolutions, used for form and texture perception. For example, when people type: these pathways are responsible for mediating the feel of keyboard keys under a moving finger and the configurations of finger joints, and for mediating the visual detection of the characters on the keys. Both vision and touch also have *transient pathways* with low spatial resolution and high temporal resolution, used for perception of motion, somatosensory flutter, and visual flicker. For example, when people dress in the morning: they feel the light presence and the fine texture of their clothing, and see the movement of the cloth, through these pathways.

Another low-level similarity is the physiological response of visual and tactile receptive fields. Legge [47] measured the sensitivity functions of visual receptive fields. In agreement with other researchers, his function appears as a wave, with both excitatory and inhibitory areas (see Figure 1.3). The wave has a large amplitude in the middle, and tapers off away from the centre. Caelli and Moraglia [15] studied the visual detection and discrimination of "optimally detectable signals". What are these signals? The Gabor function describes both these signals and Legge's sensitivity function.

This function also appears in touch. Scheibert et al. [78] conduct an experiment involving a simulated finger. They placed a pressure sensor between a cylinder and a layer of simulated skin, trying both smooth skin and fingerprinted skin. They moved the prosthetic finger across tactile textures, and measured the response of the sensor. They noted that,

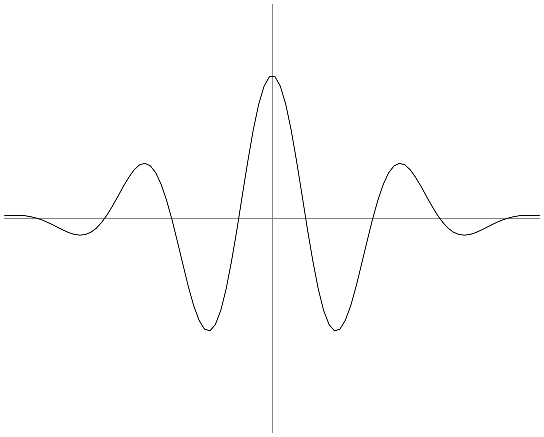


Figure 1.3: The 1D Gabor function

"Remarkably, the response function of the fingerprinted system . . . is analogous to a Gabor filter because it provides both spatial and spectral resolution." [78, page 1506]

Why is the Gabor function special? Daugman [18] lists several filters, along with their joint resolution in the spatial and frequency domains. It turns out that the Gabor function, when used as a filter, optimizes the joint resolution. Perceptual systems thus make use of approximations to Gabor filters. We can use this to our advantage to visually convey tactile textures. The tactile system responds in "units" of Gabor functions, and the visual system detects in units of Gabor functions; thus we visually communicate the properties of tactile textures with Gabor functions.

Similarities in vision and touch continue higher up in the nervous system, with possibly shared internal representations. Loomis and Klatzky [52] describe studies that support the existence of underlying modality-independent representations of objects in the central nervous system. For example, in one study children were presented with 3D shapes through either vision or touch, and later asked to examine them in the other modality. Recognition performance was 75% when visually recognizing shapes that were originally presented haptically, and 90% when haptically recognizing shapes that were originally presented visually. This is comparable to the 91% and 100% performance achieved within modalities, when presentation and later examination occurred in the same modality. This transfer of recognition implies that the nervous system connects the internal representation of haptic information to visual stimuli. This supports the idea of displaying haptic information through vision.

Besides evidence of similarities between the two modalities, there is also evidence that people can use vision to compensate for missing somatosensory feedback. Deafferented individuals are those who have lost the use of afferent nerve fibres, meaning that they are unable to receive input (from *afferent* fibres), but are still able to control their bodies (through *efferent* fibres). Two individuals of note are I. W. and G. L., neither of whom has a working somatosensory system [17]. This means that they can not feel temperature, pain, pressure, or vibration. They lack proprioception, and so do not know where their limbs are if they are not looking at them. They can not even tell how much they are exerting their muscles.

Yet I. W. and G. L. are able to stand, sit, walk, and use their hands. To be sure, they can only accomplish tasks by constantly looking at their limbs and concentrating on their actions. But the fact that they can accomplish such tasks at all means that they are able to use vision to compensate for the absence of somatosensory feedback.

In addition, Pavani and Castiello [66] conducted a study relevant to this work. Specifically, they showed a connection between shadows and tactile attention. Subjects received tactile stimulation from actuators placed on their hands, while they were shown a shadow of one of their hands on a surface. Figure 1.4 shows the actuators (filled circles) on the hands, and LEDs (hollow circles) aligned with the shadow. The shadow appeared on the surface so that the locations of the LEDs on the surface, relative to the shadow, corresponded to locations of actuators on the hand. An LED would light up, and one of the actuators would turn on. The LEDs were shown to cause interference when subjects were asked to report where they were receiving tactile stimulation. That is, reaction time and error rate increased when the position of the lit LED on the shadow did not correspond to the position of the active actuator on the hand. This study suggests that shadows can cause a *binding* between personal and extrapersonal space. Other studies by the same group [21, 67] came to the same conclusion. The latter shows that even a faked hand shadow has an effect on tactile attention, as long as the subject feels that they have control over it. This applies to our feedback mechanism because we are using a shadow-like representation of a manually-controlled probe to convey tactile information.

These connections between vision and touch all indicate that we can display haptic information through vision. What is this information? The following section describes

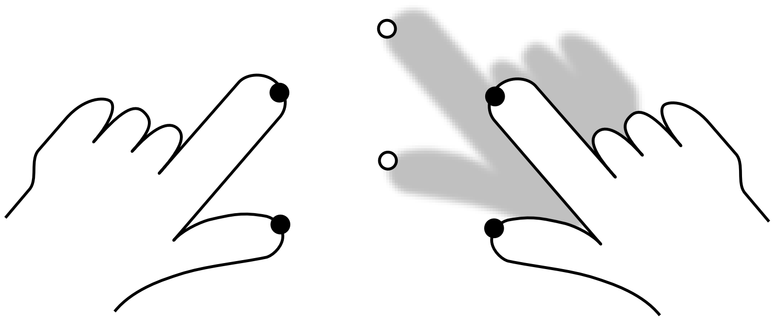


Figure 1.4: A shadow binding experiment, in which the hollow circles are LEDs on a table, and the filled circles are tactile actuators

how people feel tactile textures, and lists the important properties of these textures.

1.3 Tactile Texture Perception

How fingers feel physical texture depends on the size and spacing of the elements of these textures [34]. Skin uses spatial properties to detect large features (the order of millimetres or greater), while detection of fine features (the order of tenths of millimetres or less) relies on vibration. An example of large features are the ridges on corduroy; spatial differences in pressure on the skin lead to sensation of the ridges. It is easy to display larger features in a virtual environment, because people can directly see them, and, if desired, we can use pseudo-haptics[1] to display those features large enough to exert normal forces on a virtual probe.

Thus, this thesis focuses on the display of fine texture features, which depend on element width, element height, element spacing (density), degree of anisotropy, and (if applicable) direction. Anisotropy is the degree to which the elements point in a given direction. The feel of a completely anisotropic texture depends on direction of movement across the surface, while a completely isotropic texture is one that feels the same regardless of direction. Combed hair, for example, is made up of small, anisotropic features. These are the properties of small-feature textures that affect tactile texture perception.

Texture perception does not perceive individual elements of fine-feature textures, so it is not obviously intuitive to simply enlarge the features of the physical texture when

---

[1] See Section 2.1.

displaying it as a virtual texture. On the other hand, it does provide a convenient way to illustrate the properties of fine-feature textures.

In addition to the static properties mentioned above, our fingerprints actually produce an effect that depends on the velocity of our fingers across a surface. When a person moves a finger across a surface, fingerprints, the surface texture elements, and the velocity of the former relative to the latter, combine to create vibration. Since detection of fine features relies on vibration, surface texture feels different depending on how fast the person's finger moves.

Fine texture detection is mediated by Pacinian corpuscles [55], through vibrations generated when skin moves across a surface. Pacinian receptors are *rapidly adapting*, and are one of two transient mechanoreceptor pathways. Figure 1.5 shows the response of Pacinian corpuscles to vibration frequencies; they are most sensitive to vibrations around $250Hz$. Normal exploratory finger speed is approximately 10 to $15cm/s$. When our fingers move at these speeds, our fingerprints specifically amplify vibrations for detection near this frequency [78]. We exploit this relationship between sensation intensity and probe speed to render fine virtual textures.

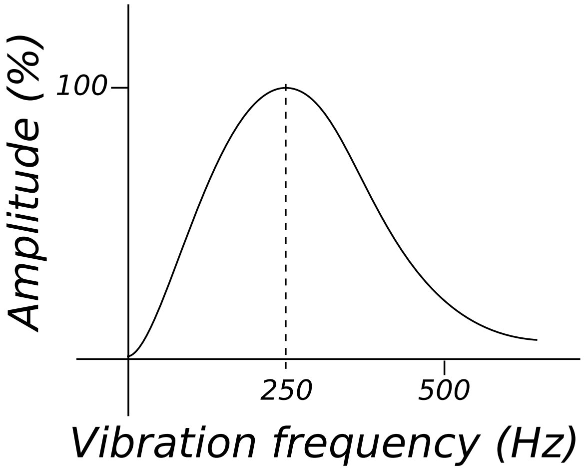


Figure 1.5: The amplitude of Pacinian corpuscle response with respect to vibration frequency

The following chapter describes work that has already been done in the area of augmenting visual feedback. Examining these feedback mechanisms reveals some major problems. The next chapter states the problems this thesis addresses, and describes the context in which we work. Chapter 4 discusses a solution to these problems, and how it was implemented. Finally, we will discuss our results, their limitations, and possible extensions.

Chapter 2

Related Work

It is possible to augment visual feedback in many different ways. See Lindeman's survey [49], for example, or Table 2.1. Most techniques display information about proximity, contact, or force (or combinations of the three). Most techniques modify the display in the neighbourhood of interaction: the probe (the virtual object through which the user interacts), the surface the user is acting upon, or the space between the two. Adding visual feedback outside this neighbourhood can be distracting. Richard and Coiffet [74], for example, report an experiment in which a separate LED display (that indicated force information) increased completion times for a manipulation task. We thus draw the pseudo-shadow in the neighbourhood of interaction.

If we do not want to modify geometry, we can use colour, transparency, or shadow approximations. For example, Ayatsuka, Matsuoka, and Rekimoto [6] describe a handheld augmented reality device that uses fake shadows to convey proximity. Each virtual object has its own light source, directly above it, that casts an umbra and penumbra onto a surface. The farther the object is from the surface, the smaller the umbra and the larger the penumbra. There are also interaction systems that substitute probes, or entire people, with just their shadows. Apperley et al. [4] dealt with two-room interaction with large displays. The display in room A represents room B's users by their shadows. Shoemaker, Tang, and Booth [79] solved the problem of interpretable interaction with large displays by projecting shadows onto them; users interact with the displays through these shadows. An example of shadows that display information specifically about the probe occurs in Ritter et al.'s [77] work. They took advantage of the two-dimensional nature of shadows

Table 2.1: Summary of feedback techniques

Technique	**Modifies**	**Displays**	**Examples**
Pseudo-haptics	probe	proximity, contact, force	[11, 46, 85, 71, 82, 89]
Transparency	probe	proximity	[14, 60, 12]
	surface	proximity, contact, force	[10, 23, 43, 80, 87]
Colour	probe, surface	proximity	[51, 50, 62, 87]
		contact	[35, 50, 23, 32, 43, 93, 19, 94, 80, 83, 41, 12, 58]
		force	[13, 51, 28, 80, 41, 81]
Shadow approximations	surface, interface	proximity	[37, 36, 13, 29, 6, 50, 4, 69, 33, 79, 22]
Symbols	interface	proximity, contact, force	[37, 36, 64, 13, 27, 74, 75, 20, 43, 53, 44, 39, 59, 61, 68, 38, 84, 26, 92, 3, 33, 80, 41, 81, 63, 22]

to display 2D visualizations and annotations directly in the shadow. Another interesting application of shadows is Herndon et al.'s [29], which does not use shadows cast by a light source. Instead it uses something more like a reflection of the probe in the surface. We base our pseudo-shadow on these ideas: representing a probe with its shadow; varying opacity to indicate distance; displaying information in the shadow itself; and removing the dependency on light sources.

We can potentially display even more information if allowed to modify probe, surface, or interface geometry. However, a problem with many techniques that seem to be highly effective is that they can not realize their potential: geometry modification might divert attention or introduce clutter [91]. The extreme of geometry modification is adding external objects or symbols to the interface between the probe and the surface. The following are two examples in which the symbols are simply lines.

Kim, Takeda, and Stark [37], for example, discuss how to improve visual feedback for remote control of a robotic arm. They render a gridded projection plane that is aligned with the arm's movement axes. The grid includes a projection of the arm, and a vertical line connecting the arm to the projection. Aguerreche, Duval, and Lécuyer [1] introduce an interaction technique for collaborative virtual environments. Their probe is a 3D pointer (an arrow), and they draw coloured balls at manipulation points on the surface. Users are able to manipulate the 3D object through the manipulation points. There is a virtual rubber band between the pointer and the manipulation point, that changes colour depending on how long it gets.

Sreng et al. [80] offer more complex examples of external objects. They focused on virtual assembly and maintenance tasks. Proximity arrows illustrate proximity between surfaces, varying in size and colour with distance, while effort arrows illustrate force information. A hybrid sphere object conveys proximity like the arrow, and compresses and bulges to show forces upon contact. Symbols and extra objects can be defined to have as many properties as desired, but again, they are among the most unnatural forms of feedback. We do not want to distract the user with the complexity or novelty of the output.

On the other hand, some examples of more natural geometry modification, occurring on the surface, are predictive dimpling[1] or menisci near the probe, and bulging out in

---

[1] William Cowan, 2009, Personal communication.

other areas of the surface when the surface is squeezed[2]. Shadows or cursors may be considered external objects, but shadows are among the most natural forms of feedback. Sreng et al. [81] introduce a friction-display technique that draws a "pencil" mark on the surface, following the probe-surface contact point. Force determines the line colour, and speed determines the line thickness. Both of these examples, shadows and pencil marks, occur directly on the surface, and both appear to be relatively natural. Feedback in the interaction interface, though, appears to be more unnatural.

Sreng et al.'s [81] pencil marks introduce another aspect of feedback: the timing. When do we start displaying something, and for how long do we display it? The pencil marks display information about the past, but it might be more useful to display information about the future. Ware [88] says, "It is clear that if creativity is to be supported, the medium must afford *tentative interactions*." It makes sense to give the user the ability to explore the virtual environment without modifying it, and our feedback mechanism does this.

We have pointed out the problems of spatial distraction and output complexity that can occur if the feedback adds geometry to the scene. It is also evident that feedback occurring on the surface is more natural than symbols appearing in the interface between the probe and the surface. We have seen that simple forms of output like shadows can be informative. These observations have influenced the design of the pseudo-shadow. The following sections contain previous work that has influenced other aspects of the pseudo-shadow.

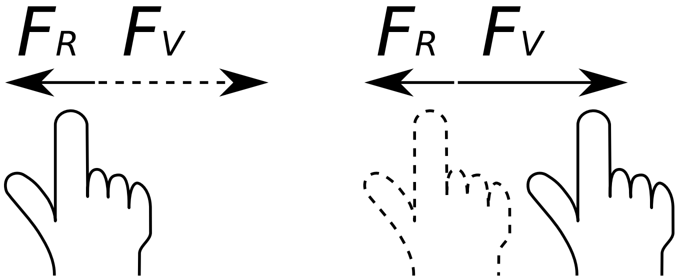
2.1 Pseudo-haptics

Pseudo-haptic feedback is another form of feedback that displays force and contact information. It modifies the control / display ratio of the probe to simulate virtual forces. See Lécuyer's survey [45]. Pusch, Martin, and Coquillart [71] conducted an experiment with a simple but effective example of pseudo-haptics. A motion capture system kept track of a user's hand; he or she also wore a head-mounted display that showed a virtual version of this hand in a virtual wind. Figure 2.1a shows the real world on the left and the virtual world on the right. Pseudo-haptics applied the force F_R that the user generated with his

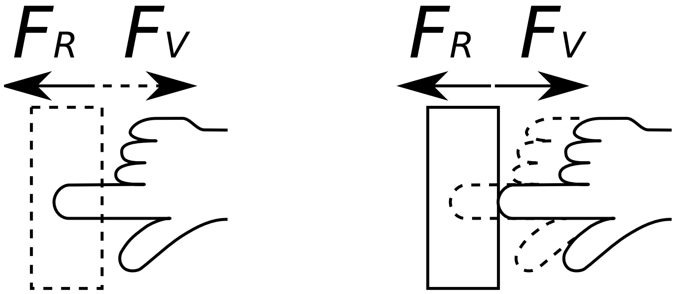
---

[2] William Cowan, 2010, Personal communication.

or her real hand (the dotted hand shows the result of this), in addition to virtual forces F_V acting on the probe in the virtual world (the solid hand shows the result of this). The dotted hand was not part of the display; it is shown here for illustrative purposes. The result was that the user saw the effects of forces applied in the virtual world, but did not sense them in the real world.



(a) A virtual wind



(b) A virtual wall

Figure 2.1: Two examples of pseudo-haptic feedback, with the real world on the left and the virtual world on the right; the user applies force F_R in the real world, while the virtual world applies both F_R and F_V

Another example of pseudo-haptics involves a virtual wall. Figure 2.1b shows the real world on the left and the virtual world on the right. The user's force alone, F_R, would push the probe into the virtual wall (shown with the dotted hand). Pseudo-haptics applies a virtual normal force F_V to the probe (the solid hand shows the result of this). In general, users exert their muscles in the real world; the virtual probe is affected by this real exertion, in addition to virtual forces from the virtual world. The user's sense of real muscle exertion, combined with a display of the virtual probe, is enough to fool users into believing the visuals from the virtual world.

This relatively simple technique substitutes visual information for proprioceptive information, and relies on the *dominance* of vision over proprioception in certain tasks. That is, if vision and proprioception receive conflicting information about the same thing, the visual information is treated as more correct.

One example of vision dominance occurs in Groen and Werkhoven's [24] study in which subjects manipulated virtual cubes. They focused on adaptation to misalignment between virtual and real hands, rather than pseudo-haptics. They found that misaligning a virtual hand is not detrimental to manipulation time or accuracy, since subjects adapt to the misalignment. Biocca, Kim, and Choi [11] studied users interacting with a virtual cadaver. Users reported feeling physical resistance, even though there was no haptic feedback; they were fooled by the visuals provided by pseudo-haptics. Steinicke et al. [82] explored the limits of virtual position and rotation displacement in the context of an immersive virtual reality environment, with the intent of enabling the virtual traversal of an infinitely large world within the confines of a finitely large real room.

Pseudo-haptics displays the effects of forces on a probe. It can display proximity information by illustrating the effects of non-contact forces that vary with distance. Contact information is displayed through the effects of normal forces. In addition, pseudo-haptic feedback is the only technique of those listed that can display tactile texture information. Watanabe and Yasumura [89] illustrate this by randomly displacing a cursor by small amounts to indicate roughness. Pseudo-haptics is limited to displaying large-feature textures, because these are the only ones that can exert forces on the probe that are large enough to cause movement. The following chapters will discuss, among other things, a solution to the problem of displaying small-feature tactile textures.

2.2 Visual Texture Synthesis

Visual texture is another way to display surface properties. The Gabor function plays a fundamental role in vision and touch (see Section 1.2), so it makes sense to use it to display tactile information visually. Porat and Zeevi [70] presented a way to synthesize textures by putting Gabor functions on a grid. They construct a lattice, and place multiple Gabor function instances at each lattice point, each instance with a different set of parameter values. By modifying the weights of each function instance, the frequency and orientation information at a given lattice point can be expressed.

About the same time, Lewis [48] introduced sparse convolution noise to generate solid textures. Lewis defined a texture as the convolution of a kernel with a set of impulses. In two dimensions, this is

$$\begin{aligned} t(x, y) &= (k * n)(x, y) \\ &= \sum_i n(x_i, y_i) k(x - x_i, y - y_i) \end{aligned}$$

in which t is the texture, k is a kernel, and n is noise defined by a set of weighted impulses. The impulses provide kernel positions (x_i, y_i) and amplitudes. The kernel parameters, if any, can be further varied based on a function of the position (x_i, y_i):

$$k(x, y) \to k_i(x, y)$$

van Wijk [86] used this idea to generate textures for data visualization. Lagae et al. [42] used the Gabor function as a kernel in sparse convolution noise. We use Lagae et al.'s work to display tactile texture information.

This chapter has listed some of the previous work relevant to this thesis. Techniques in the chapter contribute to the pseudo-shadow's implementation. This chapter has also introduced problems that appear in visual augmentation. The following chapter elaborates on these problems, and sets the context in which the pseudo-shadow solves them.

Chapter 3

Problems & Context

Each feedback technique listed in the previous chapter has its advantages, but here we focus on their problems, with the goal of developing a feedback mechanism that avoids them. The next chapter discusses what the mechanism does to solve these problems, and how it does so.

First we introduce the problem of modality appropriateness [90], which occurs when more than one modality attempts to provide information about the same thing. In the case of pseudo-haptic feedback, both vision and proprioception provide feedback about the location of the probe. The resulting percept is a mixture of the information from each modality. The modality appropriateness hypothesis states that the sense that has the greatest precision for the given task is given a greater weighting for that task. The output of the visual feedback technique must not be ignored in favour of a more reliable modality. Pseudo-haptics has the advantage that there are tasks for which vision dominates over proprioception, but what about tactile stimuli? This problem is inherently possible if we try to substitute for a modality through which a user is already getting information.

Aside from modality appropriateness, there are three other problems with existing feedback mechanisms that the pseudo-shadow addresses:

1. occlusion,
2. distraction or “unnaturalness”, and
3. lack of information content.

Because our goal is to *augment* output, we do not want to destroy, or diminish, the existing output.

Displaying additional information means either putting content on top of important existing visuals, or adding content away from the neighbourhood of interaction. Wickens and Carswell [91] discuss the problems of proximity and clutter, among others, in display design. Adding content on top of the primary visual output in the neighbourhood of interaction occludes the most important part of the primary information: the part the user is currently interacting with. On the other hand, distancing the new content results in spatially distracting the user. This is one conflict: spatial distraction versus occlusion.

Is it possible to convey as much information as arbitrary symbols do, without taking over additional screen space, or distracting the user? We can sacrifice information content to display simply-coded information in a physically small area. Alternatively, we can make the output more cognitively or perceptually complex to display a large amount of information in a physically small area. We can also display a large amount of simply-coded information in a physically large area, and sacrifice occlusion of primary information. This is another conflict: we need to minimize cognitive or perceptual distraction, lack of information content, and occlusion.

The feedback mechanism we use addresses these issues within the context outlined in the introduction. We assume the probe is any virtual object intended to be an extension of the user's hand into the virtual environment. The surface the user is interacting with should be close to flat in the neighbourhood of interaction. This enables a fast implementation of the pseudo-shadow. This is reasonable to assume, since we focus on fine textures, not large-scale textures. We also assume that we have access to standard hardware: a monitor, a graphics card, and a manual input device.

The following chapter describes how we display

- proximity and contact information,
- small-scale tactile texture information, and
- proprioceptive force information

while addressing the stated problems. Section 4.1 discusses a solution to occlusion, section 4.2 talks about distraction, section 4.3 talks about information display, and section 4.4

discusses the problem of modality appropriateness. Finally, section 4.5 describes what we do to implement the feedback technique.

Chapter 4

Technical & Implementation Details

Here we explain how we address the problems pointed out in the previous chapter. The last part of this chapter describes the implementation of the feedback mechanism.

4.1 Occlusion

A problem with displaying tactile feedback through vision is that a user usually does not see what he or she is touching. The surface is occluded by our hand, or, in a virtual environment, by the probe. There is a very simple solution: do not draw the probe. Instead, we draw only the surface, augmenting it with a shadow-like projection of the probe. This *pseudo-shadow* contains the proximity, tactile, and proprioceptive force information that we wish to convey.

The pseudo-shadow is drawn translucently. This results in less occlusion than occurs when drawing the probe. The surface might have its own visual texture, but most of this is still be visible under the transparent pseudo-shadow. Native surface texture is be completely visible outside the pseudo-shadow.

Other mechanisms have used probe transparency to alleviate this problem [14, 60, 12], but the pseudo-shadow focuses on the most important parts of the probe: those closest to the surface. The closer the probe is to the surface, the more intense the pseudo-shadow appears. We build upon many other implementations that use shadows or cursors (like Apperley et al.'s [4], or Shoemaker, Tang, and Booth's [79]). Instead of putting information about the probe into the shadow, as in Ritter et al.'s [77] work, we add information about the surface.

4.2 Distraction

Another problem is that of distraction or unnaturalness, which includes both spatial distraction, causing users to avert their visual attention, and cognitive distraction, causing the user to process unnatural information. There are natural, real-world phenomena on which people rely for information. Phenomena like shadows and interreflection are ubiquitous. People do not have to think about these phenomena. People usually use or *discount* them, automatically freeing attention for other aspects of a scene. To discount a phenomenon is to ignore its presence. These phenomena occur in areas of importance, usually without occluding important aspects of a scene.

Interreflections provide information about relative position or proximity. Shadows, in addition, help perception of surface shape and orientation[16]. The problem with shadows is that they depend on the configuration of the light sources, the caster (the probe, in our case), and the receiver (the surface, in our case). An undesirable configuration may provide no extra information, and might actually interfere with the primary information. For example, a bad shadow can hide a surface, or its boundaries can be distracting [2]. Interreflections are not very noticeable and can be difficult to render.

Madison et al. [54] discuss the effect of shadows and interreflections on a visual judgement task: deciding whether a green box is resting on a grey surface, or floating slightly above it. Shadows and interreflections were either turned off, turned on, or drawn unrealistically. The unrealistic conditions involved a white shadow, and red interreflections. Even when faked, these phenomena improved contact judgements.

Hence, these phenomena can be faked without losing their utility. People discount faked phenomena as long as they meet some basic requirements: a set of properties of the natural phenomena. Shadows have the following properties that lead to discounting:

- shadows have a gradual luminance change, as opposed to step-wise luminance changes (step-wise changes are mostly caused by reflectance changes) [9];
- moving from the edges to the centre, the shadow can not get lighter [73];

- the shadow, including the border, must be darker than the receiver [16];
- the shadow can not have reflectance changes on the border, such as marks [73] or contrast polarity reversal [16].

It is assumed that these are the properties that people most associate with shadows, and so we give these properties to our pseudo-shadow. Instead of forcing the user to discount faked phenomena, we put the above properties into the feedback technique in order to provide the same kinds of information as the natural phenomena.

Since the Gabor function occurs in both tactile and visual perceptual systems, we will use them to visually display tactile texture properties. We map the important features of small-scale tactile texture to parameters in the Gabor function. Thus the mapping is not complex or novel, and so the user is not heavily burdened with the task of decoding the output. Also, the pseudo-shadow gives us drawing space in the interaction neighbourhood, so attention is not diverted away from this area. The next section discusses how we build the procedural texture out of Gabor functions.

4.3 Information Content

We want the pseudo-shadow to evoke proximity, tactile texture, and force. We first describe how to add tactile texture information to the pseudo-shadow, and then how to add proximity and force information.

4.3.1 Tactile Texture Information

We use a kernel-convolution-based procedural texture [48, 86] to provide tactile information. Figure 4.1 shows the 2D Gabor function; we build the procedural texture out of instances of this function. Lagae et al. [42] describe how to modify the parameters of Gabor kernels. The Gabor function provides many input parameters, and the input parameters intuitively map to the visual textures we want to create. Also, the Gabor functions occupy a relatively large perceptual space, which means that people can visually distinguish between many different Gabor function instances. This makes sense, since low level human vision processes input by subjecting it to Gabor filters [47, 18].

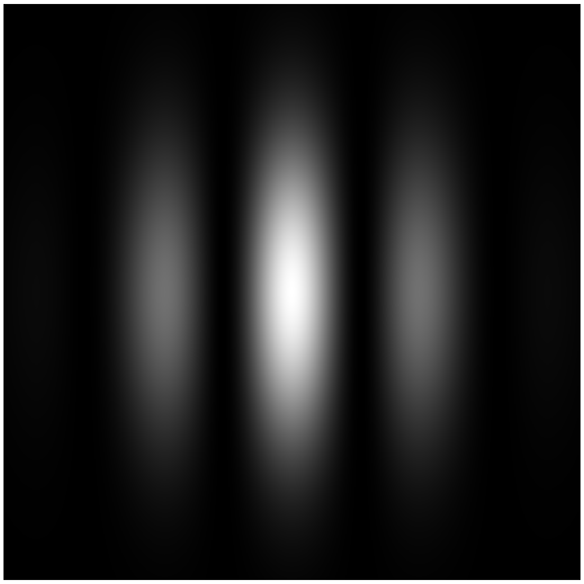


Figure 4.1: The 2D Gabor function

The basic two-dimensional Gabor function is

$$g(x,y) = \left(g_{avg} + A\cos\left(\frac{2\pi x}{\lambda} + \psi\right)\right) e^{-\frac{x^2+y^2}{W^2}}$$

in which

- (x, y) is the point of evaluation,
- g_{avg} is the average intensity,
- A is half the height of g (or the “amplitude” of g, or the contrast),
- λ is the wavelength,
- ψ is a phase shift, and
- W defines how fast the function falls off.

Adding a counter-clockwise rotation of θ, the equation becomes

$$\begin{pmatrix} x_\theta \\ y_\theta \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

$$g(x,y) = \left(g_{avg} + A\cos\left(\frac{2\pi x_\theta}{\lambda} + \psi\right)\right) e^{-\dfrac{x^2+y^2}{W^2}}$$

The important real-world properties of a surface's tactile texture are

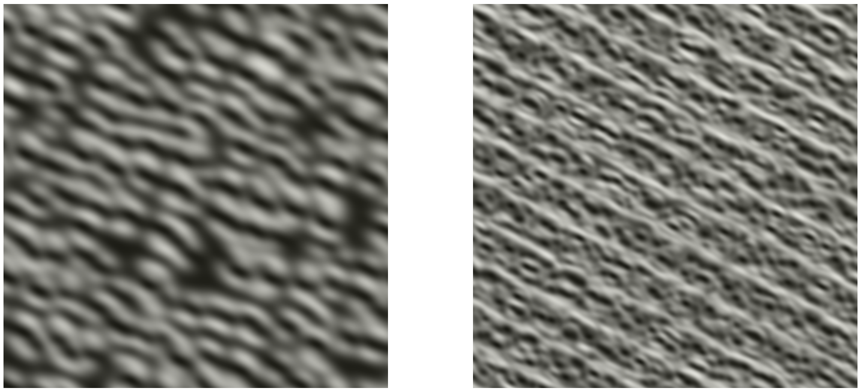
- element width,
- element height,
- element spacing, and
- element anisotropy and direction.

These are easily shown using just two Gabor kernel parameters, and two placement parameters. The kernels are placed randomly according to ranges of values of kernel densities $[n_{min}, n_{max}]$, corresponding to element spacing, and orientations $[\theta_{min}, \theta_{max}]$, allowing us to control anisotropy and direction. The texture also requires ranges of values for wavelength $[\lambda_{min}, \lambda_{max}]$, corresponding to element width, and amplitudes $[A_{min}, A_{max}]$, corresponding to element height. These eight values define the visual texture for a surface.

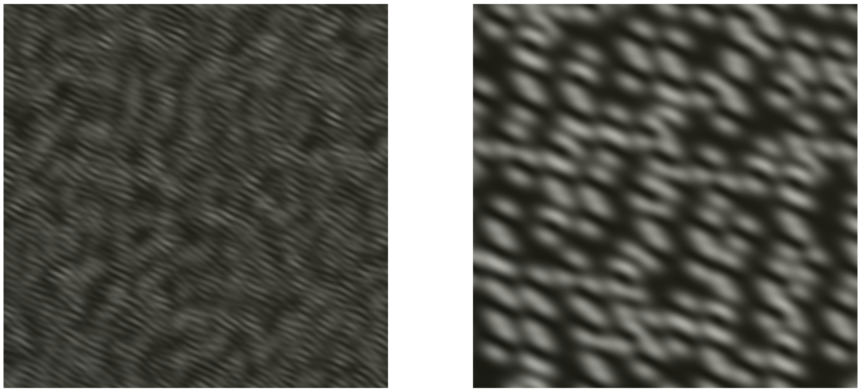
Figure 4.2 shows examples of textures generated by Gabor kernels, and Table 4.1 lists the associated parameter values. The texture elements in Figures 4.2a and 4.2d are large, while image 4.2b shows elements of varying sizes. Figure 4.2c contains elements with small height or amplitude; the elements in all other images are maximum height. Image 4.2d contains elements that are far apart, while all other images show elements close together. The texture in 4.2e is between anisotropic and isotropic, that in 4.2f is fully isotropic, and all other textures are fully anisotropic.

We further modulate the texture in two ways. There are two dynamic aspects of the interaction we wish to consider: the probe velocity and the probe-surface distance. Probe velocity contributes to the tactile texture portion of the display, so we discuss its role here. The next section discusses probe-surface distance.

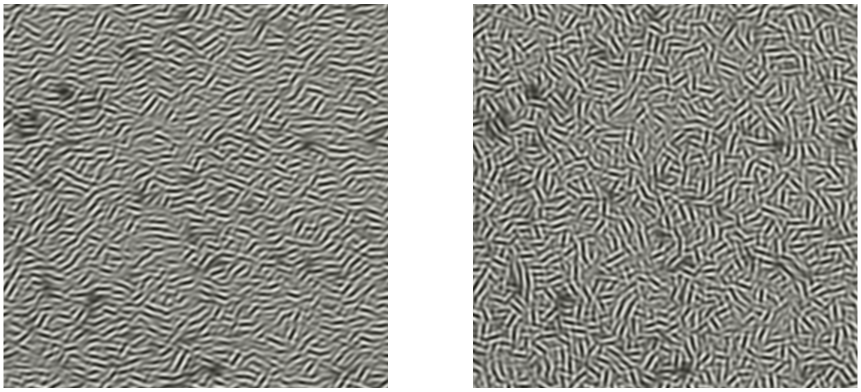
Running a real finger across a real surface produces vibration, which directly affects tactile texture perception. The probe velocity, together with the surface's defined range of wavelengths and degree of anisotropy, determine the vibration frequencies that would occur in the real world. As previously mentioned, fingerprints amplify vibrations in the



(a) Wide elements (b) Variable-width elements



(c) Short elements (d) Sparse texture



(e) Partially anisotropic texture (f) Isotropic texture

Figure 4.2: Examples of procedural textures

Table 4.1: Procedural texture parameter values for example textures

Parameter	(a)	(b)	(c)	(d)	(e)	(f)
A_{min}	1	1	0.125	1	1	1
A_{max}	1	1	0.125	1	1	1
λ_{min}	2	0.8	0.5	2	0.5	0.5
λ_{max}	2	2	0.5	2	0.5	0.5
θ_{min}	0	0	0	0	0	0
θ_{max}	0	0	0	0	$\frac{\pi}{2}$	π
n_{min}	2	100	30	1	20	20
n_{max}	6	100	30	1	30	30

$250Hz$ range, to which the Pacinian corpuscles are most sensitive. The resulting vibration frequencies correspond to how intensely a user would feel the texture, and so the probe velocity modulates the texture's amplitude according to the surface texture's wavelengths and directions.

The range of wavelengths, $[\lambda_{min}, \lambda_{max}]$, determines the optimal speed range of the probe (the range of speeds that generates $250Hz$ vibration):

$$[s_{min}, s_{max}] = [250\lambda_{min}, 250\lambda_{max}].$$

If the probe speed lies in $[s_{min}, s_{max}]$, the procedural texture amplitude should be maximized.

In general, we approximate the combination of fingerprint filtering and Pacinian response with a wide Gaussian[1], as shown in Figure 4.3. Let s be the current probe speed, and let $[f_{min}, f_{max}]$ be the range of frequencies generated by s. Then $f_{max} = \frac{s}{\lambda_{min}}$ and $f_{min} = \frac{s}{\lambda_{max}}$. The distance between the frequency range and the ideal frequency is

$$\Delta f = \begin{cases} 250 - f_{max} & \text{if } f_{max} \leq 250 \\ 0 & \text{if } f_{min} \leq 250 < f_{max} \\ f_{min} - 250 & \text{if } 250 < f_{min} \end{cases}.$$

---

[1] But see Section 5.2.

Δf attenuates the amplitude thus:

$$A_{weighted} = e^{-\frac{\Delta f^2}{w^2}} \times A.$$

Through trial and error, a value of $w = 0.008$ was found to work well. Figure 4.3 shows the effect of probe speed on texture amplitude, for a given $[\lambda_{min}, \lambda_{max}]$. Section 4.5.3 describes how we obtain the probe velocity and calculate the final texture amplitude.

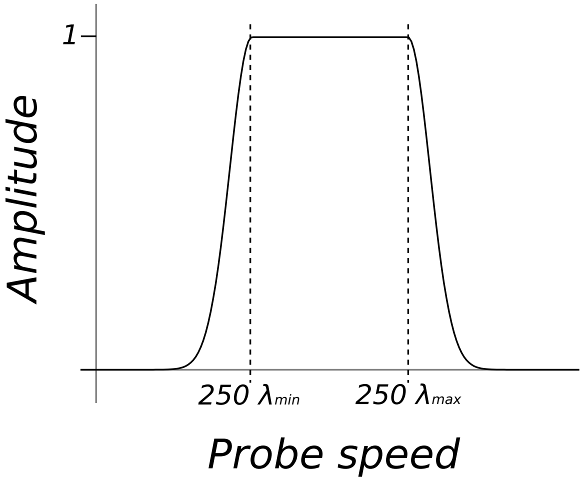


Figure 4.3: Texture amplitude modulation for a given $[\lambda_{min}, \lambda_{max}]$

4.3.2 Proximity & Contact Information

In order to convey probe-surface proximity, we use this pseudo-shadow intensity (or opacity) function:

$$intensity = \begin{cases} 1 & \text{if } distance \leq 0 \\ 0.1 + \dfrac{0.9}{distance + 1.25} & \text{otherwise} \end{cases}.$$

See Figure 4.4. Where the probe is close to the surface, the pseudo-shadow appears more intense, and where the probe-surface distance is large, the pseudo-shadow is more transparent. This modulation is similar to that of McNeely, Puterbaugh, and Troy [62], except that they illustrate surface-to-surface distance with hue.

The intensity modulation function in Figure 4.4 contains two important points. When the probe-surface distance is less than a certain threshold, the procedural texture starts to show. The user thus detects the surface texture without having to actually touch it. This point reflects the “tentativeness” of tentative interactions. The closer the point is to 0, the more the user has to actually touch the surface to determine its properties. If the point is far away from 0, the user can interact with the surface without potentially modifying it.

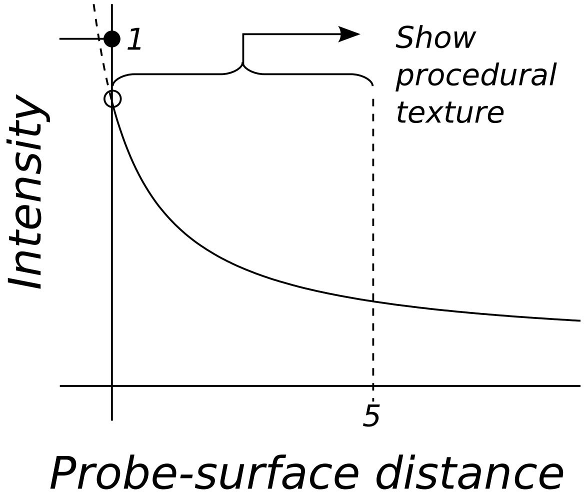


Figure 4.4: Pseudo-shadow intensity modulation

The second important point is the point of contact. Wherever the probe touches or goes through the surface, the procedural texture appears opaque. Pseudo-shadow intensity discontinuously jumps to a maximum upon contact, to clearly distinguish between *close* and *touching*.

4.3.3 Pseudo-haptics

It does not require much extra computation to add pseudo-haptic feedback, given the implementation of the rest of the feedback mechanism. The kind of proprioceptive, or force, information that we focus on is the normal force of the surface on the probe, but it is straightforward to apply the same technique to general proximity-based force fields.

The pseudo-shadow indirectly illustrates force information. As long as we apply forces to the probe when the user attempts to penetrate a surface, this will be made evident in the pseudo-shadow intensity, which conveys proximity information. When the user penetrates the surface, we displace the probe backwards to push it out of the surface. The sense of muscle effort from the user's hand or arm, and the visual output, combine to create pseudo-haptic feedback.

In addition, we can change the hardness of the surface. The speed with which the probe is pushed out of the surface gives the user a sense of how hard or soft the surface is. The faster the probe moves out of the surface, the harder the surface feels, and the slower the probe moves out, the softer the surface feels. Since the pseudo-shadow itself displays the effects of pseudo-haptic force application, we only need to solve the problem of how to actually determine and apply this force. Section 4.5.4 describes how we do this.

4.4 Modality Appropriateness

There is another problem that potentially manifests itself whenever we attempt to substitute for a modality through which a user is already getting information. For example, if a user gets touch information from a monitor while handling a physical object, like touching a mouse, is the feeling of the mouse going to completely dominate over any potential sensory substitution from the virtual environment? This is not so much a problem for us to solve, as a consideration that bears discussion. It turns out that the problem solves itself.

There are two places in which this problem can appear. Signals from touching a manual input device can potentially dominate any tactile perception owing to sensory substitution. If the user is always touching the device, though, adaptation occurs within a few minutes [56], because Pacinian mechanoreceptors are rapidly adapting. If the user is not touching the device, there is no tactile information to conflict with the visual input.

Another problem area is the conflict between the natural proprioceptive feedback a user receives when using a manual input device and the visual feedback that the pseudo-shadow provides. This problem is applicable to all other instances of pseudo-haptic feedback. Luckily, vision dominates proprioception [24], and in fact pseudo-haptics benefits from the sense of muscle exertion that accompanies using an input device. When people feel that they are exerting muscles, and they see a virtual probe standing still, they interpret this as a force blocking the probe [45]. In this way, users actually benefit from the feedback provided by the real world.

The previous sections have described the pseudo-shadow's design, and how it visually augments output while avoiding occlusion and distraction. The next section will describe the details of the pseudo-shadow's implementation.

4.5 Implementation

An important part of the pseudo-shadow is a local planar approximation of the surface. Since we assume the surface is relatively flat in the neighbourhood of interaction, we can construct a *reference plane* to approximate the local surface. See Figure 4.5. The reference plane will serve a few purposes, such as distance (d) computation and velocity (v) computation. Part of the implementation is maintaining this plane's orientation.

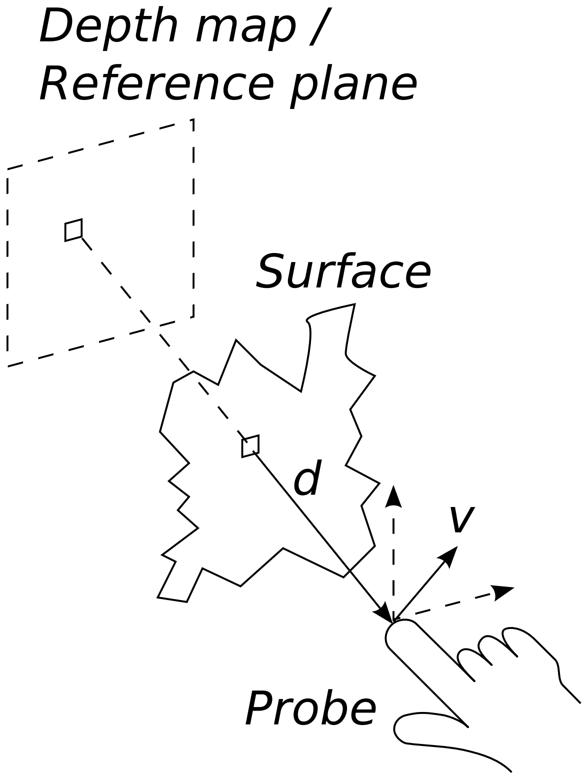
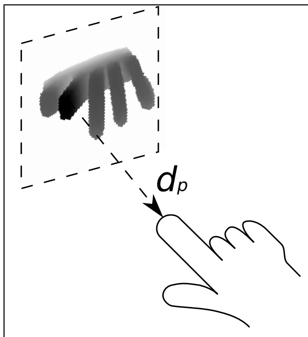
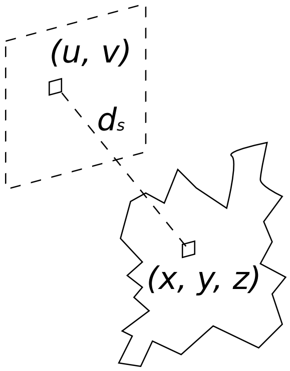


Figure 4.5: Probe-surface distance d and velocity v; the distance is for a given fragment on the surface; the probe velocity is parallel to the reference plane

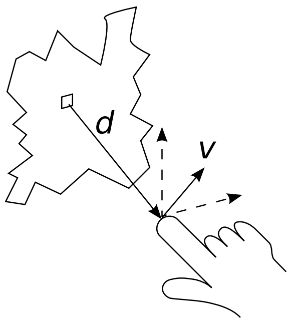
Rendering the pseudo-shadow requires the steps listed in Table 4.2. Each frame requires two renderings: an orthogonal depth map rendering of the probe, and a rendering of the



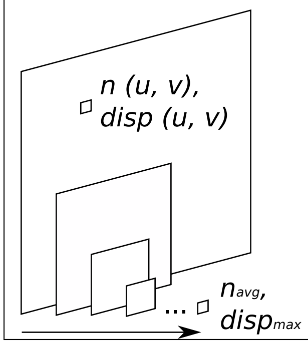
1. Align the reference plane with the surface.
2. Make a depth-map which contains the orthogonal distance d_p of the probe from the reference plane.
3. For each surface fragment,



   (a) calculate the procedural texture value;
   (b) find the surface depth d_s relative to the reference plane;
   (c) look up the probe depth d_p in the depth map;
   (d) modulate the texture amplitude using the probe velocity v and the surface anisotropy;



   (e) modulate the shadow intensity with the probe-surface distance d;
   (f) store the surface normal $n(u, v)$; and
   (g) store a pseudo-haptic displacement $disp(u, v)$ if the probe penetrates the surface.



4. Find the average surface normal n_{avg}.
5. Calculate the pseudo-haptic displacement $disp_{max}$.
6. Apply the displacement to the probe.

Table 4.2: Pseudo-shadow rendering

surface. Following the lead of Lagae et al. [42], who perform Step 3a on the GPU, we can shift onto the GPU the following computation:

- texture generation,
- distance computation,
- surface normal averaging, and
- pseudo-haptic displacement calculation.

The following sections describe these in more detail.

4.5.1 Languages and Libraries

Most of the implementation is done in C++, using OpenGL[2] and GTK+[3]. Portions of the implementation that run on the GPU are programmed in the OpenGL Shading Language (GLSL).

4.5.2 Reference Plane & Depth Map

We align the reference plane with the visible portion of the surface. This provides the best orientation for texture, distance, and velocity computation. The reference plane is initialized to point directly out of the screen. We assume that the surface is relatively flat in the neighbourhood of interaction. More precisely, we assume that most of the local surface can be represented as a height map from the reference plane, and that most of the local surface is approximately parallel to the reference plane. The first assumption allows us to calculate probe-surface distances using a depth-map. The second assumption allows us to approximate the probe velocity on the surface. Both assumptions allow us to quickly compute pseudo-haptic forces.

We align the reference plane using the currently visible surface normals. Each time we render the surface, the associated fragment shader stores each fragment's surface normal

---

[2] OpenGL 3.2 is used, with GLee managing extensions.

[3] We use gtkglextmm, in particular.

in a draw buffer (Step 3f). Step 4 then uses automatic mipmap generation, treating the draw buffer as the bottom level of a texture pyramid. See Figure 4.6. The highest level of the pyramid is thus the average of the visible surface normals. This average then becomes the reference plane's new normal. In this way the reference plane is always approximately parallel to the surface in the neighbourhood of interaction.

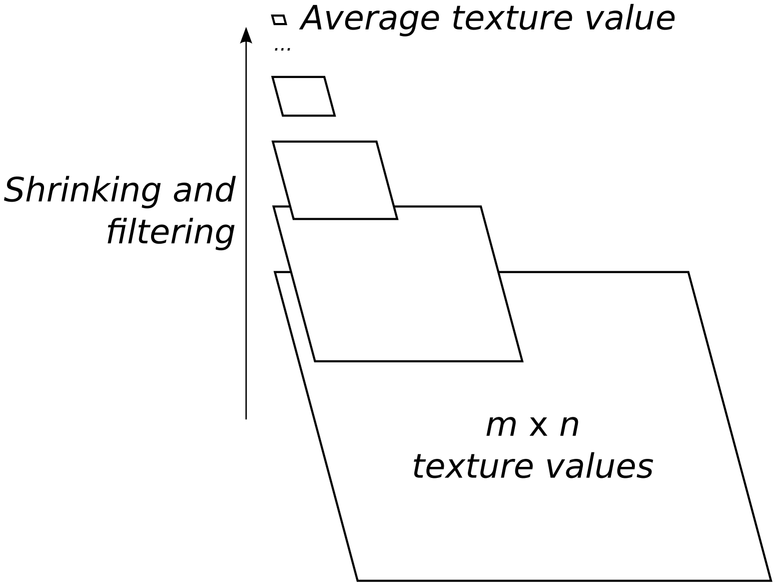


Figure 4.6: Automatic mipmapping creates each level of the texture pyramid by halving the dimensions of the texture below it, and filtering; the highest level is the average of the values in the lowest level

Each frame requires a depth map of the probe. Each time we render the probe's depth map, the corresponding fragment shader orthogonally projects the probe onto the reference plane. The shader then stores the depth values in a depth map texture. Each time we render the surface, another fragment shader needs to deal with the surface fragments. Given a point (x, y, z) on the surface, we orthogonally project the point onto the reference plane. See Figure 4.7. The reference plane has its own local coordinate frame, and the location (u, v) of the projection with respect to this frame becomes a lookup index for both the depth map and the procedural texture map. When placing Gabor kernels in the procedural texture, the texture generator uniformly samples each parameter value in its respective range. It uses the random number generator suggested by Lagae et al. [42]:

$$r_{n+1} = (3039177861 r_n) \mod 2^{32}$$

This creates a procedural texture composed of many Gabor kernels; we now need to modulate the texture.

Step 3c uses the depth map texture to find the distance d between the surface and the probe. This distance then modulates the intensity of the pseudo-shadow. The result is a blend of the procedural texture, the rest of the pseudo-shadow (which is simply black), and the underlying surface (rendered with its own lighting, textures, etc.).

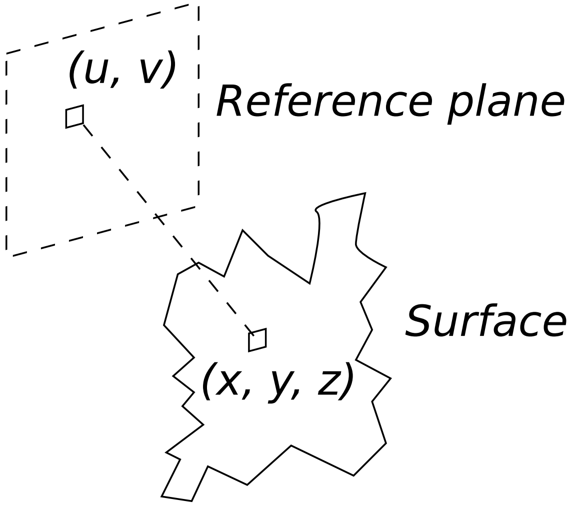
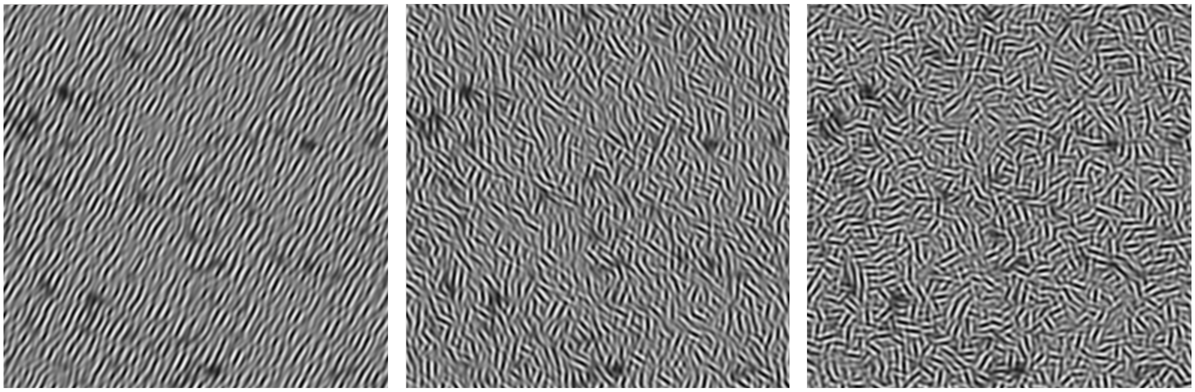


Figure 4.7: The coordinates (u, v) on the reference plane represent the surface fragment at (x, y, z)

This is similar to a reflection of the probe on the surface, and not like a shadow, in that the pseudo-shadow does not depend on a light source. In fact, it is similar to some implementations of fake shadow feedback (Ayatsuka, Matsuoka, and Rekimoto [6] or Herndon et al. [29], for example). It is unlike these implementations in part because it does not depend on the viewer, except that the pseudo-shadow fades as the surface curves away from the user's viewpoint. We assume that users are primarily interested in surfaces that are facing them. Another difference from the previous implementations is that the velocity component v of the probe, parallel to the texture variation direction, is used to modulate the texture.

4.5.3 Probe Velocity

Because a surface's tactile texture can be anisotropic or isotropic, we calculate probe speed in two ways, each producing different texture amplitude results. The degree of anisotropy of the texture then serves as a blend value between the two results. Figure 4.8 shows three levels of anisotropy, from completely isotropic to completely anisotropic. First, we use the component of the probe velocity parallel to the average texture variation direction. This makes sense if a surface texture is completely anisotropic. As textures become more isotropic, we give more weight to the probe speed in the plane parallel to the reference plane (since the plane is approximately aligned with the surface).



(a) $\theta_{max} - \theta_{min} = 0$ (b) $\theta_{max} - \theta_{min} = \frac{\pi}{2}$ (c) $\theta_{max} - \theta_{min} = \pi$

Figure 4.8: Different levels of anisotropy

If the surface texture is completely anisotropic, then $\theta_{min} = \theta_{max}$; this angle defines an axis that is parallel to the texture variation direction. (If $\theta_{min} \neq \theta_{max}$, we define this axis by averaging θ_{min} and θ_{max}.) Then the probe speed is the magnitude of the probe velocity projected onto this axis. When $\theta_{max} - \theta_{min} = \pi$, the texture is completely isotropic. In this case, we simply use the magnitude of the probe velocity projected onto the reference plane. For intermediate textures $\theta_{max} - \theta_{min}$ can be any value in $[0, \pi]$, and the final amplitude is a blend of the two extremal amplitudes. $\theta_{max} - \theta_{min}$ is the blend parameter for a simple linear blend.

Note that in order to get better velocity results from the mouse, we use the GTK+ X event filter mechanism[4], which allows us to capture X motion events to process them

---

[4] GDK::Window::add_filter

directly. Velocities were too erratic when using the GTK+ motion event handling mechanism.

4.5.4 Pseudo-haptic Feedback

The availability of a depth map substantially reduces the cost of computing the forces for pseudo-haptic feedback. The pseudo-haptic force is really a displacement that, slowly or quickly, pushes the probe back onto the surface when penetration occurs. We use the probe-surface distance and surface normals to do this. Ideally, we would calculate displacement as follows.

1. For each surface fragment,
    (a) find the probe-surface penetration,
    (b) determine this fragment's *contact region*, and
    (c) update the maximum displacement for the contact region.
2. Sum the maximum displacements from each contact region.

A *contact region* is a contiguous region of the surface that the probe currently penetrates.

In the interests of speed however, we actually calculate displacement as follows. For a given surface fragment,

$$\begin{aligned} displacement &= resistance \times -distance \times normal \\ weight &= (1 - distance)^c \\ displacement_{weighted} &= weight \times displacement \end{aligned}$$

Each local surface normal is multiplied by the probe-surface distance, and a resistance value. This becomes the displacement for the given fragment: the amount it would need to be pushed to exit the surface. The resistance is a property of the surface, used by the user to specify the surface's hardness. The displacement is then polynomially weighted with the probe-surface distance, with $c = 20$.

Step 3g stores $displacement_{weighted}$ values, and the *weight* values, each in a draw buffer. Step 5 then uses automatic mipmap generation on the $displacement_{weighted}$ and *weight* textures to find $\underset{(x,y)}{\operatorname{average}}(displacement_{weighted_{(x,y)}})$ and $\underset{(x,y)}{\operatorname{average}}(weight_{(x,y)})$. We use a large exponent to polynomially weight the *displacement* values, so that the weighted average approximates a max() operation on the *displacement* values:

$$\begin{aligned}\max_{(x,y)}(displacement_{(x,y)}) &\approx \frac{\sum\limits_{(x,y)} displacement_{weighted_{(x,y)}}}{\sum\limits_{(x,y)} weight_{(x,y)}} \\ &= \frac{\underset{(x,y)}{\operatorname{average}}(displacement_{weighted_{(x,y)}})}{\underset{(x,y)}{\operatorname{average}}(weight_{(x,y)})}\end{aligned}$$

Smaller values of the exponent c result in greater penetration, even when $resistance = 1$. The maximum displacement is the displacement that pushes the probe to the surface. Applying this displacement to the probe and rendering the new position creates pseudo-haptics.

We have shown how to visually augment probe-surface interaction in a non-detrimental manner. The pseudo-shadow avoids the problems of occlusion and distraction while displaying tactile texture, probe-surface proximity, and pseudo-haptic forces. Previous work has not described how to render small-scale tactile texture, and this is one contribution of this thesis. In addition, the implementation contains a novel way to apply pseudo-haptics using a depth-map. Also, the pseudo-shadow itself is a novel way to display probe-surface interaction. The next chapter discusses how the pseudo-shadow can be used and improved upon.

Chapter 5

Results & Discussion

The pseudo-shadow illustrates a solution to the problems listed in Chapter 3.

1. The lack of a probe, together with the transparent nature of the pseudo-shadow, overcome the problem of occlusion. A user is able to view almost the entire geometry of the surface, which is always equal to or more than what is viewable when the probe is displayed.

2. All the normally non-visual surface properties are displayed directly on the surface. This avoids the problem of dividing the user's spatial attention. We also leverage the presence of Gabor functions in both natural touch and vision to create a less complex and less novel form of output, which is expected to demand less processing.

3. In addition, Gabor functions have the potential to display a large amount of information, both in the number of variables and the resolution of each variable.

Figure 5.1 shows a hand-like probe close to a flat surface, showing only the probe-surface distance in the pseudo-shadow. Note that, since the pseudo-shadow is simply black, it is difficult to determine the boundary of the surface of contact. This is easier to see in the subsequent figure.

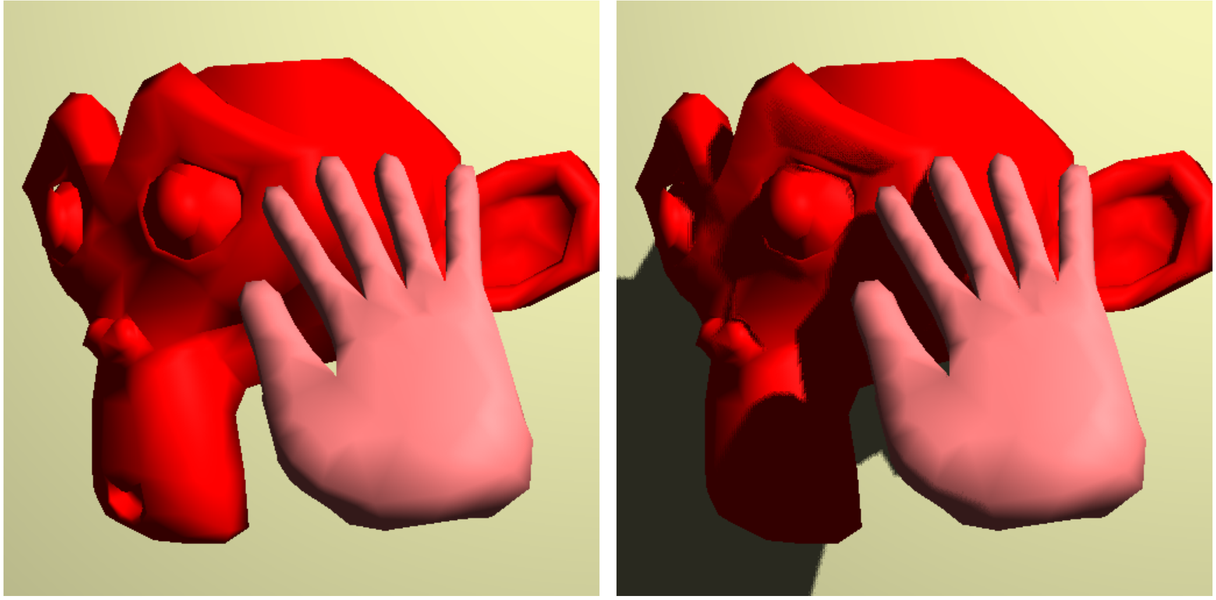
Figure 5.2 shows the hand probe interacting with a surface (a monkey head), using four kinds of feedback. There is no extra feedback in 5.2a, only the hand and the head. Figure 5.2b adds shadows, which obscure the surface while adding little information, while

Figure 5.1: Distance with no texture

Figure 5.2c shows the probe-surface distance. Figure 5.2d uses a pseudo-shadow with the procedural texture. Here the boundary between opaque and transparent pseudo-shadow is more clear, which indicates clearly where the probe is touching or penetrating the surface. Figure 5.3 contains the probe interacting with a piece of wood, rendered separately with shadows, probe-surface distance, and the pseudo-shadow. Figure 5.4 shows the same forms of feedback, this time with a piece of construction paper on the left and a piece of sandpaper on the right. Table 5.1 contains the parameter values for these surfaces.

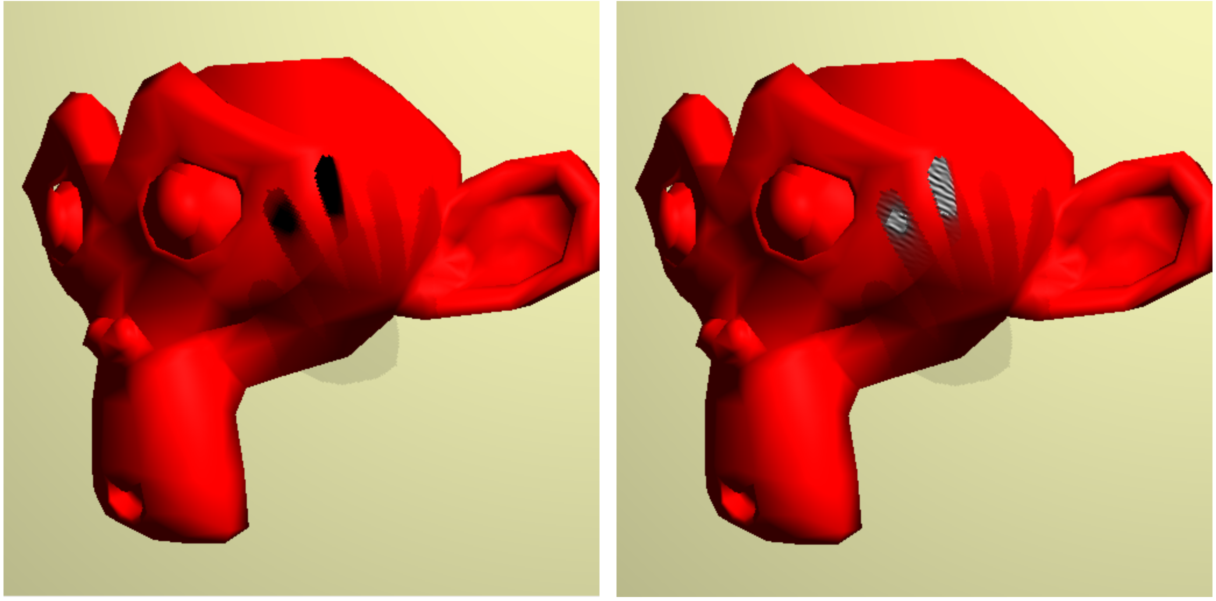
Table 5.1: Procedural texture parameter values for example surfaces

Parameter	Monkey Head	Wood	Construction Paper	Sandpaper
A_{min}	1	0.25	0.1	1
A_{max}	1	0.75	0.1	1
λ_{min}	0.5	0.1	0.25	1
λ_{max}	0.5	0.5	0.5	1.5
θ_{min}	0	0	0	0
θ_{max}	0	$\frac{\pi}{6}$	π	π
n_{min}	20	20	100	10
n_{max}	30	30	100	10



(a) No extra feedback

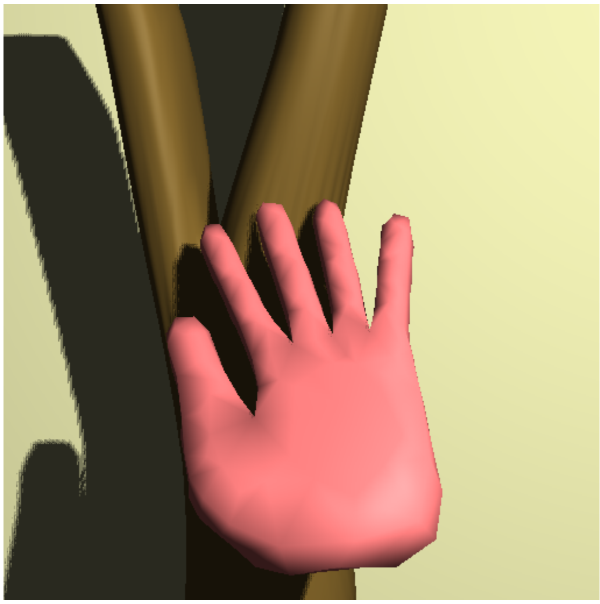
(b) A shadow



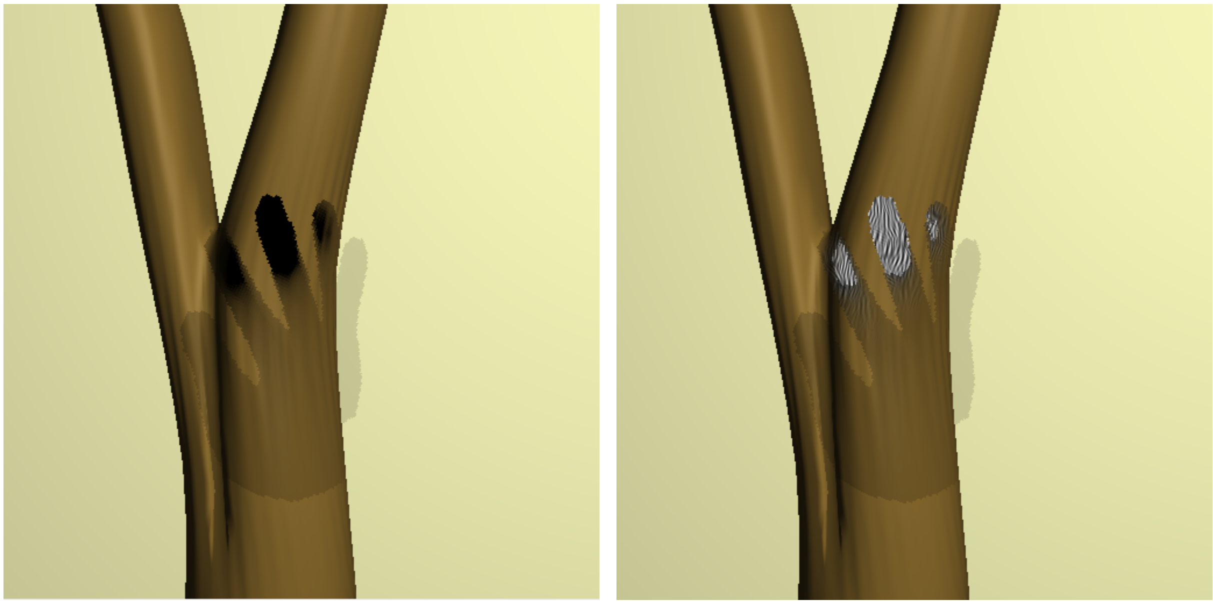
(c) Probe-surface distance

(d) A pseudo-shadow

Figure 5.2: A hand-like probe interacting with a monkey-head surface, with four kinds of feedback

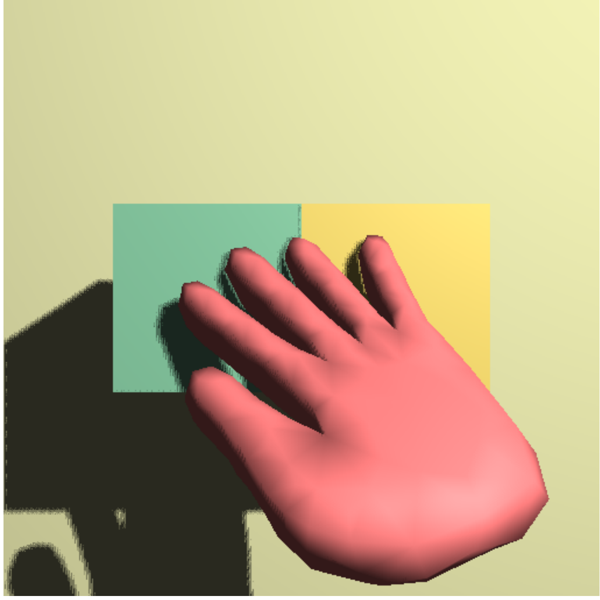


(a) A shadow

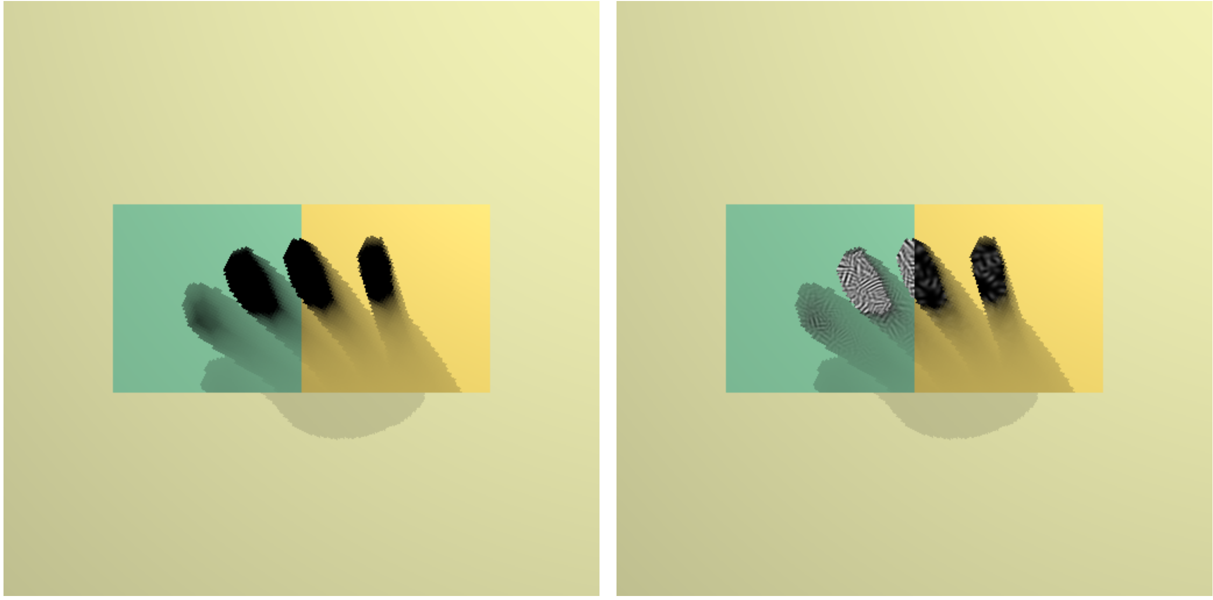


(b) Probe-surface distance (c) A pseudo-shadow

Figure 5.3: A hand-like probe interacting with a wood surface, with three kinds of feedback



(a) A shadow



(b) Probe-surface distance

(c) A pseudo-shadow

Figure 5.4: A hand-like probe interacting with construction paper (left side) and sandpaper (right side), with three kinds of feedback

Rendering the pseudo-shadow is dominated by rendering the procedural texture. Lagae et al. [42] report frame times of around $0.03s$ on a GeForce GTX 280. Procedural texture generation relies primarily on the GPU. Frame times increase to $0.04s$ or less when rendering the entire pseudo-shadow, which includes a little more work on the CPU: a Core 2 Quad Q6600.

In the fragment shader, only the portions of the surfaces near the probe are affected. Also, only the portions of the surfaces facing the user are affected. Switching surface materials requires passing to the shader only the few procedural texture parameters for that surface. This means that the cost of displaying more than one kind of surface material is negligible. The slowest pseudo-shadow computation is averaging through automatic mipmapping. This is an iterative process, in which the number of steps is logarithmic in the size of the base texture. Reducing the resolution of the base textures results in faster rendering, but lower quality. The pseudo-shadow is fast enough to be displayed on multiple surfaces during real-time interaction, with good quality.

5.1 Limitations

One reason for introducing pseudo-haptics into the feedback is that it requires little extra work. In calculating pseudo-haptic displacement, mipmapping approximates the maximum penetration of the probe into the surface. This displacement "maximum" is not actually a maximum: it is an average. It is thus always smaller than the actual maximum displacement required to push the probe completely out of the surface. This means that we can not achieve completely hard surfaces; even if $resistance = 1$, there will always be some penetration. But we can also use standard, and possibly slower, collision detection and handling for this.

If we want to further improve collision detection and handling with more directions (a different direction for each of a set of surfaces), the GPU needs to do a little more work. Each new direction, corresponding to a section of approximately flat surfaces, requires a new depth map. Each new depth map requires rendering from an additional reference plane for each frame. Each of these renderings, in turn, requires mipmapping, which is already an inherently iterative process. Hence, in order to improve collision detection and handling, we need to create and keep track of more reference planes. It might make sense to add a small constant number of reference planes, but more would greatly increase frame times.

This feedback technique is intended for larger virtual surfaces, or at least objects that users look at and touch in the same areas simultaneously. The latter is not necessarily true when manipulating small objects (those users can pick up) in the real world. When someone looks at a small 3D object, they tend to get to know the front (the side facing them) of it better than the back; when someone feels a small 3D object, they get to know the back of it better than the front [52]. Any kind of visual feedback is limited to being displayed on the visible surface of the object.

Another problem is the range of tactile textures we can represent. In theory, we could concurrently display any number of tactile textures. We would need to re-map the tactile texture element properties into our space of Gabor properties. This introduces a perceptual problem, though. We would need to experimentally determine the actual resolution of each Gabor property in order to determine how many tactile textures a user could distinguish between.

5.2 Extensions & Future Work

We approximate the relationship between tactile texture intensity and probe speed with a Gaussian:

$$A_{weighted} = e^{-\frac{\Delta f^2}{w^2}} \times A.$$

We could, however, interpolate values in a texture look-up. After experimentally deriving a fine enough sample of perceived intensities for given frequencies, it would be straightforward to put them into a 1D texture and perform linear interpolation. Experimentally varying the relative direction of probe motion, and the range of directions $\theta_{max} - \theta_{min}$, would result in a 3D texture, from which we could once again easily interpolate values. This would result in a better mapping from probe velocity to visual texture amplitude.

Another need for experimentally derived data occurs in determining what visualizations correspond to particular properties we want to convey. In fact, it is already known that our sensory modalities are not completely independent, even across different sense organs. A stimulus in one modality can be associated with a completely different stimulus in another modality. For example, Marks [56] describes a study in which users were asked to associate a sound's loudness with an object's warmth. This is not a subjective phenomenon, but is substantiated by replicable experimental results. Inter-modal associations do not stop at artificial questions asked of subjects. Certain stimuli can independently cause evocations in other modalities.

An extreme case of this is synesthesia. Unfortunately for us, synesthesia of the "minor" senses (taste, smell, touch, temperature, pain) is idiosyncratic, and non-synesthetic equivalence might be more idiosyncratic, and less rigid [57]. But we might still be able to use equivalence classes of stimuli to strengthen a particular feeling we are trying to display.

Stimuli often share hedonic attributes, such as discomfort or pleasantness. For example, small changes in skin temperature, warmth, and quiet sounds are more pleasurable than no changes or large changes in temperature, cold, and loud sounds [56]. Stimuli can also have common dimensions, like extent, intensity, and "brightness" [57]. Some examples of bright phenomena are cold, sting, weak pain, smooth, hard, sharp, and light, while some examples of dull phenomena are warmth, pressure, rough, soft, blunt, and heavy. It is thus likely possible to simulate extra modalities in order to strengthen a primary sensation.

Going beyond the original context of the problem, Gabor textures have many parameters that a user can manipulate. It thus makes sense to use Gabor function parameters to display other tactile surface properties, and even non-tactile properties [86]. This feedback mechanism provides a canvas in which to put information of any kind, as long as it corresponds to exploratory or manipulative probe procedures.

Going beyond vision, audition has been shown to interact with tactile perception. Kruijff et al. [40] constructed a pen-input device that provides audio and tactile feedback. They report on results of an object placement task, finding that audio-tactile feedback can improve collision detection and texture perception, and that audio feedback was more important than vibrotactile feedback for texture recognition. Guest et al. [25] show how audio feedback can change subjects' roughness perception: amplifying high-frequency audio content tends to make a surface feel rougher, while attenuating high-frequency content tends to make a surface feel smoother. With the addition of sounds, we might be able to display more information, or to display information more powerfully.

In addition to extending the feedback mechanism, there is a question of whether a user actually "feels" the virtual surface. Or do users have to actively think about how the visual output corresponds to tactile properties? Bach-y-Rita [8, page 98] relates an anecdote involving the Tactile Visual Sensory Substitution (TVSS) system. The TVSS presents the image from a camera on a tactile stimulator on the user's back. A blind TVSS user was connected to the system and controlling the attached camera. The experimenter changed the zoom level, which is normally under the user's control, causing the camera to zoom in. The user then flinched backward and raised his arms, afraid that the objects he was "looking" at were flying towards him, despite receiving the camera display on his back. This response indicates how natural the user finds TVSS feedback after adequate training. Would a visual pin-prick in the pseudo-shadow cause an experienced user to jerk back a hand?

How long would it take to learn to associate the pseudo-shadow's visual output with forces, tactile textures, and proximity? We could ask a subject to first physically examine a set of real-world surfaces, and then visually interact with the same set of surfaces in a virtual world. Would the user be able to recognize the surfaces in the virtual world? Another experimental question is: does the pseudo-shadow improve the effectiveness or efficiency of a user? Would the pseudo-shadow benefit the user in a high-level task, such as 3D modelling?

Chapter 6

Conclusion

This work has investigated augmenting visual feedback in interactive virtual environments. We sought visually to depict in the virtual world stimuli that are primarily non-visual in the real world. In particular, we have outlined some of the current techniques for displaying proximity, tactile, and proprioceptive information, along with some of their larger problems. We have focused on manual interaction in 3D virtual environments, in which the user is using a manual input device. We defined a set of small-scale tactile texture properties, together with distance and force, to display. Our constraints required us to display multiple dimensions of content, while being as non-distracting as possible, and not reducing the information content of the primary visual information.

The pseudo-shadow meets these constraints. For tactile textures it displays element spacing, direction, width, and height; it also displays probe-surface distance and contact information, showing the effects of pseudo-haptics. Since the GPU can do a great deal of the work, the pseudo-shadow renders fast enough for interaction with multiple surfaces.

In this work, we have provided the following contributions.

- We use a depth map and texture mipmapping for a simple collision detection and handling mechanism. This mechanism can generalize to applying forces caused by distance-based force fields.
- The pseudo-shadow is a method for displaying content during probe-surface interaction that reduces occlusion and spatial distraction.

- We use a visual texture based on Gabor kernels, combined with a modulation based on probe velocity, for visual display of small-scale tactile texture.

It would be interesting to test the pseudo-shadow's effectiveness in information display in a formal experiment. If users are successful in getting a sense of "feeling" the associated tactile textures, what other modalities might the pseudo-shadow be able to display? The procedural texture of Lagae et al. [42] can display many variables, and the pseudo-shadow provides surface space for this procedural texture. To what degree can sensory substitution be used to put the real world into a virtual one?

References

[1] Laurent Aguerreche, Thierry Duval, and Anatole Lécuyer. 3-hand manipulation of virtual objects. In Michitaka Hirose, Dieter Schmalstieg, Chadwick A. Wingrave, and Kunihiro Nishimura, editors, *EGVE/ICAT/EuroVR*, pages 153–156. Eurographics Association, 2009. 13

[2] D. Akers, F. Losasso, J. Klingner, M. Agrawala, J. Rick, and P. Hanrahan. Conveying shape and features with image-based relighting. In *IEEE Visualization, 2003*, pages 349 –354, Oct 2003. 22

[3] T. Akinbiyi, C. E. Reiley, S. Saha, D. Burschka, C. J. Hasser, D. D. Yuh, and A. M. Okamura. Dynamic augmented reality for sensory substitution in robot-assisted surgical systems. In *28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2006*, pages 567 –570, Aug 2006. 12

[4] Mark Apperley, Laurie McLeod, Masood Masoodian, Lance Paine, Malcolm Phillips, Bill Rogers, and Kirsten Thomson. Use of video shadow for small group interaction awareness on a large interactive display surface. In *Proceedings of the Fourth Australasian user interface conference on User interfaces 2003 - Volume 18*, AUIC '03, pages 81–90, Darlinghurst, Australia, Australia, 2003. Australian Computer Society, Inc. 11, 12, 21

[5] Roland Arsenault and Colin Ware. Eye-hand co-ordination with force feedback. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '00, pages 408–414, New York, NY, USA, 2000. 2

[6] Yuji Ayatsuka, Satoshi Matsuoka, and Jun Rekimoto. Penumbrae for 3d interactions. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*, UIST '96, pages 165–166, New York, NY, USA, 1996. 11, 12, 35

[7] P. Bach-y Rita, C. C. Collins, F. Sauders, B. White, and L. Scadden. Vision substitution by tactile image projection. *Nature*, 221:963–964, 1969. 2, 4

[8] Paul Bach-y Rita. *Brain mechanisms in sensory substitution.* Academic Press, New York, 1972. 4, 5, 47

[9] S. S. Bergstroem, K. A. Gustafsson, and T. Jakobsson. Distinctness of perceived three-dimensional form induced by modulated illumination: effects of certain display and modulation conditions. *Perception & Psychophysics*, 53(6):648–657, 1993. 22

[10] Eric A. Bier, Maureen C. Stone, Ken Pier, William Buxton, and Tony D. DeRose. Toolglass and magic lenses: the see-through interface. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '93, pages 73–80, New York, NY, USA, 1993. 12

[11] Frank Biocca, Jin Kim, and Yung Choi. Visual touch in virtual environments: An exploratory study of presence, multimodal interfaces, and cross-modal sensory illusions. *Presence*, 10(3):247–265, 2001. 12, 16

[12] A. Bloomfield and N. I. Badler. Collision awareness using vibrotactile arrays. In *IEEE Virtual Reality Conference, 2007*, pages 163 –170, March 2007. 12, 21

[13] Frederick P. Brooks, Ming Ouhyoung, James J. Batter, and P. Jerome Kilpatrick. Project grope– haptic displays for scientific visualization. *ACM Computer Graphics*, 24(4):177–185, 1990. SIGGRAPH 1990 Conference Proceedings. 12

[14] Volkert Buchmann, Trond Nilsen, and Mark Billinghurst. Interaction with partially transparent hands and objects. In *Proceedings of the Sixth Australasian conference on User interface - Volume 40*, AUIC '05, pages 17–20, Darlinghurst, Australia, Australia, 2005. Australian Computer Society, Inc. 12, 21

[15] T. Caelli and G. Moraglia. On the detection of gabor signals and discrimination of gabor textures. *Vision Research*, 25(5):671–684, Nov 1985. 5

[16] P. Cavanagh and Y. G. Leclerc. Shape from shadows. *Journal of Experimental Psychology: Human Perception & Performance*, 15:3–27, 1989. 22, 23

[17] J. Cole and J. Paillard. Living without touch and peripheral information about body position and movement: Studies with deafferented subjects. In A. J. Marcel J. L. Bermudez, editor, *The body and the self*, pages 245–266. MIT Press, Cambridge, MA, 1995. 7

[18] J. G. Daugman. Spatial visual channels in the fourier plane. *Vision Research*, 24(9):891–910, Jan 1984. 6, 23

[19] W. Edwards, Woodrow Barfield, and A. Nussbaum. The use of force feedback and auditory cues for performance of an assembly task in an immersive virtual environment. *Virtual Real.*, 7:112–119, April 2004. 12

[20] B. Frohlich, H. Tramberend, A. Beers, M. Agrawala, and D. Baraff. Physically-based manipulation on the responsive workbench. In *Proceedings, IEEE Virtual Reality 2000*, pages 5 –11, 2000. 12

[21] Giovanni Galfano and Francesco Pavani. Long-lasting capture of tactile attention by body shadows. *Experimental Brain Research*, 166:518–527, 2005. 7

[22] Michael Glueck, Keenan Crane, Sean Anderson, Andres Rutnik, and Azam Khan. Multiscale 3d reference visualization. In *Proceedings of the 2009 symposium on Interactive 3D graphics and games*, I3D '09, pages 225–232, New York, NY, USA, 2009. 12

[23] Antonino Gomes de Sá and Gabriel Zachmann. Virtual reality as a tool for verification of assembly and maintenance processes. *Computers & Graphics*, 23(3):389 –403, 1999. 12

[24] J. Groen and P. J. Werkhoven. Visuomotor adaptation to virtual hand position in interactive virtual environments. *Presence-Teleoperators and Virtual Environments*, 7(5):429–446, Oct 1998. 16, 30

[25] Steve Guest, Caroline Catmur, Donna Lloyd, and Charles Spence. Audiotactile interactions in roughness perception. *Experimental Brain Research*, 146(2):161–171, 2002. 46

[26] M. Hamdi and A. Ferreira. Microassembly planning using physical-based models in virtual environment. In *Proceedings, IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004*, volume 4, pages 3369 – 3374 vol.4, Sept 2004. 12

[27] Claudia Hendrix and Woodrow Barfield. Relationship between monocular and binocular depth cues for judgements of spatial information and spatial instrument design. *Displays*, 16:103–113, January 1995. 12

[28] I. Herbst and J. Stark. Comparing force magnitudes by means of vibro-tactile, auditory, and visual feedback. In *IEEE International Workshop on Haptic Audio Visual Environments and their Applications, 2005*, page 5 pp., Oct 2005. 12

[29] Kenneth P. Herndon, Robert C. Zeleznik, Daniel C. Robbins, D. Brookshire Conner, Scott S. Snibbe, and Andries van Dam. Interactive shadows. In *Proceedings of the 5th annual ACM symposium on User interface software and technology*, UIST '92, pages 1–6, New York, NY, USA, 1992. 12, 13, 35

[30] S. S. Hsiao. Similarities between touch and vision. In J. W. Morley, editor, *Neural aspects in tactile sensation*. Elsevier, 1998. 5

[31] Helen H. Hu, Amy A. Gooch, William B. Thompson, Brian E. Smits, John J. Rieser, and Peter Shirley. Visual cues for imminent object contact in realistic virtual environment. In *Proceedings of the conference on Visualization '00*, VIS '00, pages 179–185, Los Alamitos, CA, USA, 2000. IEEE Computer Society Press. 1

[32] Brent Edward Insko. *Passive haptics significantly enhances virtual environments*. PhD thesis, The University of North Carolina at Chapel Hill, 2001. AAI3007820. 12

[33] Volkert Jurgens, Andy Cockburn, and Mark Billinghurst. Depth cues for augmented reality stakeout. In *Proceedings of the 7th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction: design centered HCI*, CHINZ '06, pages 117–124, New York, NY, USA, 2006. 12

[34] D. Katz. *The world of touch*. Erlbaum, Hillsdale, NJ, 1989. Translated by Krueger, L. E. Original work published in 1925. 8

[35] W. S. Kim and A. K. Bejczy. Graphics displays for operator aid in telemanipulation. In *Conference Proceedings, IEEE International Conference on Systems, Man, and*

Cybernetics, 1991, Decision Aiding for Complex Systems, pages 1059 –1067 vol.2, Oct 1991. 12

[36] W. S. Kim and L. W. Stark. Cooperative control of visual displays for telemanipulation. In *Proceedings, IEEE International Conference on Robotics and Automation, 1989*, pages 1327 –1332 vol.3, May 1989. 12

[37] W. S. Kim, M. Takeda, and L. Stark. On-the-screen visual enhancements for a telerobotic vision system. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 1988*, volume 1, pages 126 –130, August 1988. 12, 13

[38] M. Kitagawa, D. Dokko, A. M. Okamura, and D. D. Yuh. Effect of sensory substitution on suture-manipulation forces for robotic surgical systems. *Journal of Thoracic and Vardiovascular Surgery*, 129(1):151–158, 2005. 12

[39] Michal Koutek, Jeroen van Hees, Frits H. Post, and A. F. Bakker. Virtual spring manipulators for particle steering in molecular dynamics on the responsive workbench. In *Proceedings of the workshop on Virtual environments 2002*, EGVE '02, pages 53–ff, Aire-la-Ville, Switzerland, Switzerland, 2002. Eurographics Association. 12

[40] Ernst Kruijff, Gerold Wesche, Kai Riege, Gernot Goebbels, Martijn Kunstman, and Dieter Schmalstieg. Tactylus, a pen-input device exploring audiotactile sensory binding. In *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '06, pages 312–315, New York, NY, USA, 2006. 46

[41] Gregorij Kurillo, Matja Mihelj, Marko Munih, and Tadej Bajd. Multi-fingered grasping and manipulation in virtual environments using an isometric finger device. *Presence: Teleoperators and Virtual Environments*, 16(3):293–306, 2007. 12

[42] Ares Lagae, Sylvain Lefebvre, George Drettakis, and Philip Dutré. Procedural noise using sparse gabor convolution. *ACM Trans. Graph.*, 28:54:1–54:10, July 2009. 17, 23, 33, 34, 44, 49

[43] A. Lécuyer, A. Kheddar, S. Coquillart, L. Graux, and P. Coiffet. A haptic prototype for the simulations of aeronautics mounting/unmounting operations. In *Proceedings 10th IEEE International Workshop on Robot and Human Interactive Communication, 2001*, pages 182 –187, 2001. 12

[44] A. Lécuyer, C. Megard, J.-M. Burkhardt, T. Lim, S Coquillart, P Coiffet, and L. Graux. The effect of haptic, visual and auditory feedback on an insertion task on a 2-screen work-bench. In *Proceedings of the Immersive Projection Technology (IPT) Symposium*, 2002. 12

[45] Anatole Lécuyer. Simulating haptic feedback using vision: A survey of research and applications of pseudo-haptic feedback. *Presence: Teleoperators and Virtual Environments*, 18(1):39–53, 2009. 14, 30

[46] Anatole Lécuyer, Jean-Marie Burkhardt, and Chee-Hian Tan. A study of the modification of the speed and size of the cursor for simulating pseudo-haptic bumps and holes. *ACM Trans. Appl. Percept.*, 5:14:1–14:21, September 2008. 12

[47] G. E. Legge. Space domain properties of a spatial frequency channel in human vision. *Vision Research*, 18:959–969, 1978. 5, 23

[48] J. P. Lewis. Algorithms for solid noise synthesis. *SIGGRAPH Comput. Graph.*, 23:263–270, July 1989. 17, 23

[49] Robert W. Lindeman. Virtual contact: the continuum from purely visual to purely physical. In *In Proceedings of the 47th annual meeting of the human factor and ergonomics society*, pages 2103–2107, 2003. 11

[50] Robert W. Lindeman, John L. Sibert, and James K. Hahn. Towards usable vr: an empirical study of user interfaces for immersive virtual environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems: the CHI is the limit*, CHI '99, pages 64–71, New York, NY, USA, 1999. 12

[51] Robert William Lindeman. *Bimanual interaction, passive-haptic feedback, 3d widget representation, and simulated surface constraints for interaction in immersive virtual environments.* PhD thesis, The George Washington University, 1999. AAI9923558. 12

[52] Jack M. Loomis and Roberta L. Klatzky. Functional equivalence of spatial representations from vision, touch and hearing: Relevance for sensory substitution. In J. J. Rieser and A. L. Corn, editors, *Blindness and brain plasticity in navigation and object perception.* Lawrence Erlbaum Associates, 2007. 6, 45

[53] E. B. Lum, A. Stompel, and Kwan-Liu Ma. Kinetic visualization: a technique for illustrating 3d shape and structure. In *IEEE Visualization, 2002*, pages 435 –442, Nov 2002. 12

[54] Cindee Madison, Daniel J. Kersten, William B. Thompson, Peter Shirley, and Brian Smits. The use of subtle illumination cues for human judgement of spatial layout. Technical Report UUCS-99-001, Computer Science Department, University of Utah, January 1999. 22

[55] J. C. Makous, R. M. Friedman, and C. J. Vierck. A critical band filter in touch. *The Journal of Neuroscience*, 15(4):2808–2818, 1995. 9

[56] Lawrence E. Marks. Temporal factors in sensation. In *Sensory Processes: the New Psychophysics*. Academic Press, 1974. 30, 46

[57] L.E. Marks. *The unity of the senses: interrelations among the modalities*. Academic press series in cognition and perception /ed. by Edward Carterette, Morton P. Friedman. Academic Press, 1978. 46

[58] J. Martin, J. Savall, I. Diaz, J. Hernantes, and D. Borro. Evaluation of sensory substitution to simplify the mechanical design of a haptic wrist. In *IEEE International Workshop on Haptic Audio visual Environments and Games, 2008*, pages 132 –136, Oct 2008. 12

[59] A. H. Mason and C. L. MacKenzie. The effects of visual information about self-movement on grasp forces when receiving objects in an augmented environment. In *Proceedings, 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2002*, pages 105 –112, 2002. 12

[60] Andrea H. Mason. An experimental study on the role of graphical information about hand movement when interacting with objects in virtual reality environments. *Interacting with Computers*, 19(3):370 – 381, 2007. 12, 21

[61] Andrea H. Mason and Christine L. Mackenzie. The role of graphical feedback about self-movement when receiving objects in an augmented environment. *Presence: Teleoper. Virtual Environ.*, 13:507–519, October 2004. 12

[62] W. A. McNeely, K. D. Puterbaugh, and J. J. Troy. Voxel-based 6-dof haptic rendering improvements. *Haptics-e*, 3(7), 2006. 12, 28

[63] Guillaume Millet, Anatole Lécuyer, Jean-Marie Burkhardt, D. Sinan Haliyo, and Stéphane Régnier. Improving perception and understanding of nanoscale phenomena using haptics and visual analogy. In *Proceedings of the 6th international conference on Haptics: Perception, Devices and Scenarios*, EuroHaptics '08, pages 847–856, Berlin, Heidelberg, 2008. Springer-Verlag. 12

[64] O.-Y. Ming, D. V. Beard, and Jr. Brooks, F. P. Force display performs better than visual display in a simple 6-d docking task. In *Proceedings, IEEE International Conference on Robotics and Automation, 1989*, pages 1462 –1466 vol.3, May 1989. 12

[65] Saskia K Nagel, Christine Carl, Tobias Kringe, Robert Mrtin, and Peter Knig. Beyond sensory substitution—learning the sixth sense. *Journal of Neural Engineering*, 2(4):R13, 2005. 4

[66] Francesco Pavani and Umberto Castiello. Binding personal and extrapersonal space through body shadows. *Nature Neuroscience*, 7(1):14–6, 2004. 7

[67] Francesco Pavani and Giovanni Galfano. Self-attributed body-shadows modulate tactile attention. *Cognition*, 104(1):73 – 88, 2007. 7

[68] Bernd Petzold, Michael F. Zaeh, Berthold Faerber, Barbara Deml, Hans Egermeier, Johannes Schilp, and Stella Clarke. A study on visual, auditory, and haptic feedback for assembly tasks. *Presence: Teleoperators and Virtual Environments*, 13(1):16–21, 2004. 12

[69] Wayne Piekarski. *Interactive 3d modelling in outdoor augmented reality worlds.* PhD thesis, University of South Australia, 2004. 12

[70] M. Porat and Y.Y. Zeevi. Localized texture processing in vision: analysis and synthesis in the gaborian space. *IEEE Transactions on Biomedical Engineering*, 36(1):115 –129, Jan 1989. 16

[71] A. Pusch, O. Martin, and S. Coquillart. Hemp-hand-displacement-based pseudo-haptics: A study of a force field application. In *IEEE Symposium on 3D User Interfaces, 2008*, pages 59 –66, March 2008. 12, 14

[72] C. M. Reed, N. I. Durlach, and L. D. Braida. Research on tactile communication of speech: a review. *ASHA Monogr*, 20:1–23, 1982. 4

[73] Ronald A Rensink and Patrick Cavanagh. The influence of cast shadows on visual search. *Perception*, 33(11):1339–1358, 2004. 22, 23

[74] P. Richard and P. Coiffet. Human perceptual issues in virtual environments: sensory substitution and information redundancy. In *Proceedings, 4th IEEE International Workshop on Robot and Human Communication, 1995*, pages 301 –306, Jul 1995. 11, 12

[75] P. Richard and P. Coiffet. Dextrous haptic interaction in virtual environments: human performance evaluations. In *8th IEEE International Workshop on Robot and Human Interaction, 1999*, pages 315 –320, 1999. 12

[76] R. J. Riso. Sensory augmentation for enhanced control of fns systems. In A. Mital, editor, *Ergonomics in rehabilitation*. Taylor and Francis, 1988. 4

[77] Felix Ritter, Henry Sonnet, Knut Hartmann, and Thomas Strothotte. Illustrative shadows: integrating 3d and 2d information displays. In *Proceedings of the 8th international conference on Intelligent user interfaces*, IUI '03, pages 166–173, New York, NY, USA, 2003. 11, 22

[78] J. Scheibert, S. Leurent, A. Prevost, and G. Debrgeas. The role of fingerprints in the coding of tactile information probed with a biomimetic sensor. *Science*, 323(5920):1503–1506, 2009. 5, 6, 9

[79] Garth Shoemaker, Anthony Tang, and Kellogg S. Booth. Shadow reaching: a new perspective on interaction for large displays. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*, UIST '07, pages 53–56, New York, NY, USA, 2007. 11, 12, 21

[80] J. Sreng, A. Lécuyer, C. Megard, and C. Andriot. Using visual cues of contact to improve interactive manipulation of virtual objects in industrial assembly/maintenance simulations. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1013–020, Sept.-Oct. 2006. 12, 13

[81] Jean Sreng, Florian Bergez, Jérémie Legarrec, Anatole Lécuyer, and Claude Andriot. Using an event-based approach to improve the multimodal rendering of 6dof virtual contact. In *Proceedings of the 2007 ACM symposium on Virtual reality software and technology*, VRST '07, pages 165–173, New York, NY, USA, 2007. 12, 14

[82] Frank Steinicke, Timo Ropinski, Gerd Bruder, and Klaus Hinrichs. The holodeck construction manual. In *ACM SIGGRAPH 2008 posters*, SIGGRAPH '08, pages 97:1–97:3, New York, NY, USA, 2008. 12, 16

[83] Alistair Sutcliffe, Brian Gault, Terence Fernando, and Kevin Tan. Investigating interaction in cave virtual environments. *ACM Trans. Comput.-Hum. Interact.*, 13:235–267, June 2006. 12

[84] M. Tavakoli, R. V. Patel, and M. Moallem. Robotic suturing forces in the presence of haptic feedback and sensory substitution. In *Proceedings of IEEE Conference on Control Applications, 2005*, pages 1 –6, Aug 2005. 12

[85] Koert van Mensvoort, Dik J. Hermes, and Maurice van Montfort. Usability of optically simulated haptic feedback. *International Journal of Human-Computer Studies*, 66(6):438 – 451, 2008. 12

[86] Jarke J. van Wijk. Spot noise texture synthesis for data visualization. *SIGGRAPH Comput. Graph.*, 25:309–318, July 1991. 17, 23, 46

[87] Lode Vanacken, Tovi Grossman, and Karin Coninx. Multimodal selection techniques for dense and occluded 3d virtual environments. *International Journal of Human-Computer Studies*, 67(3):237 – 255, 2009. Current trends in 3D user interface research. 12

[88] Colin Ware. *Information Visualization: Perception for Design.* Morgan Kaufmann Publishers, 2000. 14

[89] Keita Watanabe and Michiaki Yasumura. Visualhaptics: generating haptic sensation using only visual cues. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, ACE '08, pages 405–405, New York, NY, USA, 2008. 12, 16

[90] R. B. Welch and D. H. Warren. Immediate perceptual response to intersensory discrepancy. *Psychological bulletin*, 88(3):638–667, November 1980. 18

[91] C. D. Wickens and C. M. Carswell. The proximity compatibility principle: its psychological foundation and relevance to display design. *Human Factors*, 37(3):473–494, Sept 1995. 13, 19

[92] Hao Xin, C. M. Burns, and J. S. Zelek. Non-situated vibrotactile force feedback and laparoscopy performance. In *IEEE International Workshop on Haptic Audio Visual Environments and their Applications, 2006*, pages 27 –32, 2006. 1, 12

[93] Mihaela A. Zahariev and Christine L. MacKenzie. Auditory, graphical and haptic contact cues for a reach, grasp, and place task in an augmented environment. In *Proceedings of the 5th international conference on Multimodal interfaces*, ICMI '03, pages 273–276, New York, NY, USA, 2003. 12

[94] Ying Zhang, Terrence Fernando, Hannan Xiao, and Adrian R. L. Travis. Evaluation of auditory and visual feedback on task performance in a virtual assembly environment. *Presence: Teleoper. Virtual Environ.*, 15:613–626, December 2006. 12