Depth from Shading as an Attentional Cue in User Interfaces

by

Sandra Lynn Loop

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

Waterloo, Ontario, Cananda, 1995

Abstract

When using a computer, we are engaging in a dialogue with the computer. In order to communicate effectively with the computer, we must be aware of the state of the computer. Therefore, we need our attention drawn to particular areas that may indicate the state. When these areas are not found easily, that is these areas are not conspicuous, then we may sacrifice speed and accuracy resulting in potentially disastrous consequences.

It has been known for years that colour is very conspicuous: it is easily noticed and attracts attention. For this reason, colour has been used in the interface to group, discriminate and draw attention. Visual texture, normally studied for its segregating and discriminating properties, is also very conspicuous. Another visual attribute that research has shown to be conspicuous is depth conveyed through shape from shading perception: convex and concave objects are easily discriminated.

The three-dimensional look that shape from shading imparts is becoming more popular in interface design due to its concrete, realistic appeal. However, designers are not making use of the functional benefits, that is the conspicuousness, of depth from shading in the design of the interface. A windowing system is a good example of an interface that needs to present conspicuous information to the user so that the user knows the state of the computer. In this case, so it is known which window is active and can receive input. This thesis empirically supports the idea that depth from shading can be used as a conspicuous attibute in a windowing system to indicate the active window. However, as the depth from shading cue is obscured by other overlapping windows, the conspicuousness of the cue decreases. Some guidelines for maintaining conspicuity of the depth from shading cue are provided.

Acknowledgements

I would like to thank my supervisor, William Cowan, for his advice and guidance throughout this thesis and for believing in my abilities before my entrance into the Master's program. My readers, Marceli Wein, Pierre Jolicoeur, Greg Veres, and Fabrice Jaubert, deserve thanks for their wisdom and attention. Furthermore, I thank Pierre for his expertise and advice regarding the 'Attention' literature. A big thanks goes to Anne Jenson for helping things go smoothly over the past couple years.

I also appreciated and enjoyed the many talks I had with Ian Bell regarding the different stages of my thesis and for providing motivation to explore the art of academic writing. I also want to thank Glenn Paulley for his wise advice.

Greg Veres deserves much thanks for spending much time giving advice and sharing his C++ and object-oriented design expertise, and, of course, for his support and love.

Finally, I would like to thank friends and family for their love and support. In particular, I thank my family for supporting my decisions during my academic program changes throughout the last nine years.

The work in this thesis was partially supported by the University of Waterloo via a Provost's Scholarship.

Trademarks

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Human-Computer Dialogue

When using a computer, users are engaging in a dialogue with the computer. They indicate what they would like done and the computer processes the information. The computer may prompt the user for more information or may indicate in some manner that the processing is done. In order to communicate effectively with the computer, the users must be aware of the state of the computer. That is, they must know when and where the computer is able to accept information from them.

This knowledge of the state of the computer can be maintained by the user's memory given the history of actions or by visual presentation of the state. Note that in both cases, the acquisition and maintenance of this knowledge may potentially result in less efficient work on the task at hand due to the increased cognitive load. When there is more to remember and to process cognitively - when the cognitive load is increased - extra resources are required to deal with the increased load which might mean that there will be fewer resources available for other tasks. Simply stated in economic theory terms, when cognitive resources are in short supply, there is a high demand.

When the information is presented visually, users need their attention drawn to particular areas that may indicate the state of the computer. If these areas are not found easily, then they may sacrifice speed and accuracy. They may spend too much time and effort attending to the interface, searching for cues to the state. And they may incorrectly perceive the state and thus perform incorrect actions. Consider the situation where two applications (A and B) are open and each presents a window. You wish to quit application A and thus you visually focus on the window associated with this application and you type ‘quit’. Unknown to you, application B was the active application. Therefore, the quit command was executed on application B rather than application A with potentially damaging results. Thus, poor information presentation increases time, effort, and errors. The next few sections discuss factors that influence information presentation.

1.2 Cognitive Load

Cognitive load refers to the amount of information that the user must cognitively process. It is desirable to minimize the cognitive load required to perform a task because in addition to the potential sacrifice in speed and accuracy, cognitive load correlates directly with other factors such as fatigue, stress, inability to ‘timeshare’, and learning time (Scott, 1991).

Cognitive load can be affected by factors such as voluntary or involuntary control of attention. Over a century ago, Fechner (1860) observed that attention could be under voluntary or involuntary control. Voluntary cues require attention to the target characteristics and claim attention for some part of the stimulus, whereas involuntary cues demand attention but do not require it to operate. For instance, in the auditory channel, the perception of a loud noise when the level of background noise is low is involuntary: the loud noise is involuntarily attended to. Yet, the constant hum of a fan or a computer is only noticed if the listener voluntarily wants to hear it. This distinction between voluntary and involuntary control is true of the visual channel as well. A bright blinking light is attended to involuntarily, whereas colour is attended to voluntarily, that is, the observer must decide which colour to attend to.

Involuntary control of attention does not require cognitive overhead to attend to the stimulus. However, switching contexts from the involuntary stimulus back to the original task requires cognitive overhead. Thus, this type of stimulus such as a blinking light will be distracting and will increase the users' cognitive load if the users' attention is unnecessarily drawn back and forth between the stimulus and the original task.

Voluntary control of attention requires cognitive overhead to search deliberately for the stimulus as well as to switch back to the original context. However, this extra cognitive load is incurred only when the user needs to attend to the stimulus. Therefore, for most elements of a user interface, in particular when the cues are continuously present, perception of the cues through voluntary control of attention is most beneficial in terms of decreasing cognitive load. Furthermore, the more the stimulus can be perceived quickly, accurately, and effortlessly, the more the cognitive load is reduced.

1.3 Attentional Cues

The problem of requiring voluntary attentional cues exists in many different situations. A form-based interface must indicate the current input destination as well as possible alternatives. Drawing or painting software must indicate in which mode the interface is. For instance, the current mode of input could be the paintbrush or it could be the rotation tool and the interface should provide an attentional cue to indicate which mode is active. The important point is that user actions are interpreted differently according to the current context. Another common situation requiring attentional cues is that of determining the active window in windowing environments. And it is the windowing environment context to which the problem domain will be restricted.

Most computers these days use windowing systems to manage the multiple contexts involved in the tasks or subtasks for which computers are used. Multiple windows may present multiple data streams or may present multiple views on a single data stream, however, there is only one input stream. Therefore, only one window at a time can be active to receive input; which window is active may be referred to as the state of the system.

Ideally, attention is focused solely on the task - not on the process of communication with the computer or the determination of state or context. Therefore, cues are needed to reduce the amount of attention involved in determining the state of the computer. Such cues should be conspicuous: easily noticed, voluntary-attention attracting, and automatically processed.

Visual search experiments are useful for determining cue conspicuity. Search times that increase linearly with the number of elements displayed indicate inconspicuous cues since every element is being inspected individually. However, search times that do not increase with the number of elements displayed, indicate conspicuous cues that draw attention easily. Such cues ‘pop out’ so that it is unnecessary to inspect each element individually.

Contemporary windowing systems use a variety of attentional cues. For example, Motif™ uses colour as a cue. The frame of the active window is painted in a different colour than the inactive windows: the default is blue for the active window and grey for the inactive windows. The X Window System’s™ standard window manager, twm, also uses colour to differentiate the active window: the titlebar is highlighted with a different colour. The Macintosh™ windowing environment uses visual texture to highlight the active window; the active window has six horizontal bars in the titlebar, whereas the inactive windows do not. Nor do they have the close or zoom boxes in the titlebar, or the scrolls and scale box. In addition to these type of visual cues, in some systems the active window is the top window, whereas in others it is not necessarily so.

1.3.1 Colour as an Attentional Cue

Colour works well as an attentional cue. It has been known for decades that colour differences can be used to make something very conspicuous. Green and Anderson (1956) examined the degree to which colour coding facilitated search time of displayed two digit numbers. They found that when the subjects know the colour of the target beforehand, the search time is approximately proportional to the number of symbols of the target's colour. When the subjects do not know the target's colour, search time depends primarily on the total number of symbols on the display. Thus, when the colour of the target is known, and thus can be used as an attentional cue, only the items in the specified colour need attention. Smith (1962) also found that when the colour of the target is known in advance, search times are considerably shorter than when the target colour is not known. More recently researchers have found that visual search efficiency when searching for a colour target depends on the dissimilarity between the targets and distractors and the similarity between the distractors. Nagy and Sanchez (1990, 1992) found that the mean search time increases linearly with the number of distractors if chromaticity differences are small, whereas mean search time is roughly constant if chromaticity differences are large. Further, other researchers have found that visual search efficiency increases continuously with decreasing similarity between targets and distractors and increasing similarity between two distractors

(Duncan, 1989; Carter, 1982). Smallman and Boynton (1990) found that search time for a target is quick when multiple colours are displayed if they are well separated in colour space. The search times when basic nameable colours and when equally discriminable nonbasic colours are displayed are similarly quick. Thus, colour works best as an attentional cue when the distractors are similar or homogeneous and the targets are dissimilar from the distractors, or when the multiple colours of the targets and distractors are well separated in colour space.

Because of the usefulness of colour as an attentional cue, research has been conducted to assess its applicability to windowing environments. MacIntyre (1991) created a dynamic colour management system for windowing interfaces where colour is used as a cue both to group windows and to differentiate windows. In this system, windows may be grouped, for instance, by a particular attribute such as application type. Windows of different groups are assigned different colours. When a new window with a new colour is added to the display the other windows dynamically alter their colour to separate themselves in colour space while satisfying aesthetic and functional constraints. In addition, the window that is active is a more saturated colour. This system does not necessarily draw attention to the state, that is to an input area or the input window, however, it can allow the user to focus attention on a particular type or colour of window and provides a means of streaming the different contexts.

1.3.2 Visual Texture as an Attentional Cue

Visual texture discrimination has also been studied for decades using several different techniques.The low-level processes that are responsible for texture discrimination are thought to be related to the processes that cause ‘pop-out’ in a visual search task. Some approaches to study visual texture discrimination use similarity or difference rating methods (Beck, 1966a,b, 1967; Harvey & Gervais, 1982; Beck et al, 1987;

Purks & Richards, 1977; Amadasun & King, 1989) or classification (Harvey & Gervais, 1978; Rao & Lohse, 1993; Tamura et al, 1978) in which various textures are grouped in such a way that differences within groups are minimized. Another approach, called texture segregation determination, is to use rapid detection of texture differences as an indication of discrimination. In these experiments, one texture is shown embedded within another and the observer must detect its presence or its position with the briefly presented display. Using the texture segregation approach, Julesz (1981, 1986) defined the important dimensions based on his texton theory, where texture segregation is the result of the computation of the first-order statistics of the textons or conspicuous local features. The textons he found were nonoverlapping line segments, terminators (line endings), line crossings and blobs. The important information encoded in blobs is the length, width, and orientation. Beck (1966a, 1966b, 1967) claims that the properties brightness, colour, movement, size and slopes of contours, and lines of figures are important. Triesman (1985) adds to the list contrast, line curvature, and closure.

Ware and Knight (1992) are undertaking a major study into the properties of three dimensions - orientation, spatial frequency, and contrast (OSC texture space). They are trying to categorize these dimensions by presenting subjects with three textures. The subjects adjust the texture element in the middle until it appears to lie midway between the two flanking patterns. The new subdivided sections are recursively subdivided until the textures are indistinguishable. A visual texture space, once developed, would help in predicting whether two textures are differentiable. Perhaps, like the Smallman and Boynton (1990) study, search time for a target would be quick when multiple visual textures are displayed if they were well separated in visual texture space.

Lin (1993) studied empirically whether visual texture is conspicuous and could also be used as a window system attentional cue similar to colour. His first experiment presented a patch of visual texture in the centre of the display screen. This patch was then surrounded by two, four, six, or eight other patches of visual texture, where one of these patches was the same texture as that used in the centre. The subjects were to indicate which texture patch matched the centre texture patch. An experiment of similar design, except the textual names of the textures were used as stimuli, became the control condition. Lin found that reaction time was roughly constant for the visual texture condition, that is visual texture ‘pops-out’, whereas reaction time for the text condition increased linearly with the number of displayed texture names. These results suggest that since multiple textures are displayed and the target texture ‘pops out’, then visual texture could be used effectively for grouping and discriminating windows similar to MacIntyre’s use of colour.

Lin’s second experiment also presented a patch of visual texture in the centre of the display screen. This patch was then surrounded by patches of visual texture. The subject was to respond as to whether all the texture patches were the same or one was different. The control condition was again a replacement of the visual textures with their textual names. Again, he found that reaction time for the visual texture condition is roughly constant, whereas reaction time increases linearly with the number of texture names in the text condition. These results suggest that visual texture could be used to indicated the active window. Similar to presenting the active window in a particular colour and all others in a second colour, the active window could be presented in a distinctive visual texture.

1.3.3 Depth from Shading as an Attentional Cue

As discussed previously, conspicuous cues - those that can be perceived quickly, accurately, and effortlessly - serve well as attentional cues in a windowing interface. In addition to both colour and visual texture, previous research has shown that depth conveyed through shape from shading can be perceived quickly, accurately, and effortlessly. The perception of shape from shading is the ability to perceive a three-dimensional object or shape based on a shaded two-dimensional representation. Experiments have shown that subjects effortlessly distinguish concave shaded objects from convex shaded objects (Pentland, 1987; Ramachandran, 1988; Kleffner & Ramachandran, 1992; Braun, 1993), therefore, possibly making depth from shading a good candidate for an attentional cue in a windowing interface.

While it may be possible to differentiate various degrees of depth, this is not done in practice and so depth from shading has only two states (convex and concave), whereas both colour and visual texture have many usable instances. That is, multiple colours can be used simultaneously in an interface. This is true of visual textures as well. Thus, windows can be grouped and discriminated according to different criteria using these multi-dimensional attributes. Colour and visual texture can also be used to represent binary states such as active and inactive windows. Motif uses two colours to indicate active versus inactive windows, and Experiment 2 of Lin's research also addressed this situation. However, used in this manner, colour and visual texture fail to exploit their full potential. Depth from shading is not a good candidate for grouping and discriminating windows in multiple categories because depth from shading can only represent two states, but it is ideally suited for representing binary states such as active and inactive windows. Also, with a large repertoire of attentional cues available, redundancy can easily be designed into an interface. If more than one cue is presented to the user, it is more likely that the state will be perceived properly.

Many current window systems display their interfaces using shape from shading to give a concrete three-dimensional appearance. However, the designers only recognize the concrete aspect of shape from shading; they are not aware of any other possible benefits (Osborne & Thomas, 1993). Thus, even though these interfaces exhibit shape from shading, they are not designed to take advantage of the depth from shading cues.

1.4 Summary

The goal of this thesis is to determine if depth from shading can be used effectively in a windowing interface. It must be determined if the shapes commonly used to create this type of interface are perceived effortlessly. (Most of the previously reported studies in the literature use egg and egg-crate type shapes (see Figure 2-2).) Also it must be assessed whether the manner in which we use these shapes affect perception. That is, is the depth from shading cue effective when it is partially occluded by another object, such as would happen if used in overlapping windows.

Chapter 2 explores the research that has already been conducted with respect to the shape from shading phenomena. Chapter 3 details the implementation of the psychological experimentation framework. The experiments and results are presented in Chapter 4 while the discussion and future work is discussed in Chapter 5. Chapter 6 describes a demonstration application that was created to illustrate the benefits of a window system using depth from shading as an attentional cue for the active window.

Chapter 2

Background

It makes sense that humans are well-equipped for perceiving depth. It is important to see the uneven ground on which we walk and to process objects moving rapidly in depth in order to function well in our three-dimensional environment. Many cues provide such depth information: binocular disparity, convergence, stereopsis, linear perspective, aerial perspective, and so on. The perception of depth through shape from shading cues is probably one of the most primitive operations to recover the third dimension, as is evident from the evolution of pale undersides among the most primitive species of animals (Thayer, 1909). This type of countershading compensates for shading effects caused by overhead lighting such as the sun, flattens the animal's shape, and reduces its contrast with the surrounding environment: an effective means of escaping a predator's view.

It is useful to understand the properties of shape from shading in order to use them in the design of interfaces. For instance, which conditions affect the perception of shape or form and which conditions ensure a consistent perception of depth? It is also beneficial to become acquainted with the results of other research measuring the effortless processing of depth from shading in order to better understand the phenomenon. These issues are explored in this chapter as well as a suitable method to assess effortless processing and the statistical tests that are used to analyze the results.

2.1 History

It has long been known that shape is conveyed through shading. In the early 1400s, Masaccio discovered chiaroscuro (*chiaro*, or light; *scuro*, or dark), which refers to the gradations between light and dark that produce the effect of modelling, or of light reflected from three-dimensional surfaces. As illustrated in Masaccio's *The Tribute Money,* Masaccio portrays the bulk of the figures not through generalized modelling

Figure 2-1: Masaccio, *The Tribute Money,* c.1427
Masaccio's discovery of chiarascuro is illustrated in this painting. Photo printed in de la Croix/ Tansey, 1986 courtesy of Fratelli Alinari, Art Resource, New York.

with a flat, neutral light with no identifiable source, as his predecessors did, but by means of a light that comes from a specific source outside the picture, striking the figures at an angle, illuminating the parts of the solids that obstruct its path and leaving the rest in deep shadow (see Figure 2-1). This placement of light against dark gives the illusion of deep sculptural relief. The masses we see are visible only because of the direction and intensity of the light. This technique was carried on and was perfected by the masters of the High Renaissance era exemplified by the paintings of Leonardo Da Vinci, Michelangelo, and Raphael.

It has also been known for a long time that protuberances and indentations in surfaces represented through shading and viewed monocularly can be ambiguous (Rittenhouse, 1786). There are always two possible interpretations for a given pattern of shading - a convex surface apparently lit from one direction or a concave surface lit from the opposite direction. Furthermore, there is evidence that when people are misinformed about the direction of illumination, they tend to misperceive surface relief (Rittenhouse, 1786; Brewster, 1832; Oppel, 1856; Yonas et al, 1979; Berbaum et al, 1983). However, all objects are perceived consistently given a single lightsource.

2.1.1 Single Lightsource Assumption

As would be expected given a visual environment with a single main light source, the sun, most perceptual systems assume there to be a single light source. Consider the display in Figure 2-2. Circles in the left column are shaded left to right from light to dark; conversely, circles in the right column are shaded from dark to light. These three-dimensional objects can be seen either as concave or convex depending on the direction of light. If the apparent light source is from the left, the left column appears convex and the right column appears concave. The convexity and concavity reverses if the apparent light source is from the right. However, the brain has no way of determining which side the direction of light is coming from. The visual system performs a global operation which assumes a single light source to resolve the ambiguity (Ramachandran, 1988). As soon as one object is determined, all the other objects are seen with a consistent light source with respect to this object.

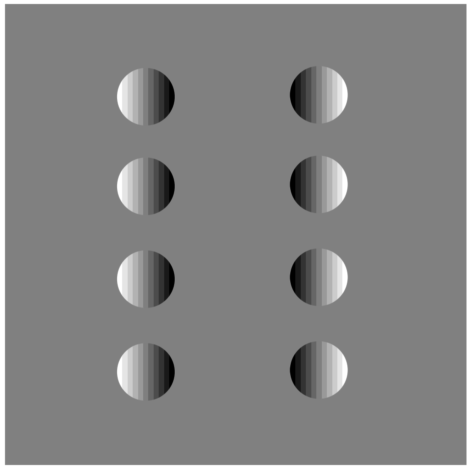


Figure 2-2: Demonstration of single lightsource assumption
Due to the biological assumption of a single lightsource, either the left column appears concave and the right column convex or vice versa.

2.1.2 Overhead Lighting Assumption

In addition to the single lightsource assumption, there is also an overhead lighting assumption. If a flat representation of a surface, such as that shown in Figure 2-3, is viewed, a region in which the light part is above the dark part (group A) tends to appear as a convexity and a region in which the dark part is above the light part (group B) tends to appear as a concavity. Brewster (1826) concluded from this fact that people resolve the ambiguity of shading by assuming that the source of illumination is from above.

Furthermore, the assumption of overhead lighting is implied by the countershading observed in animals (Thayer, 1909; Ramachandran, 1988b; Kleffner & Ramachandran, 1992). This, too, is what would be expected given the overhead light source of the sun. More specifically, the visual system assumes a retina-centric light source (Ramachandran, 1988a, 1988b; Kleffner & Ramachandran, 1992; Yonas, Kuskowshi

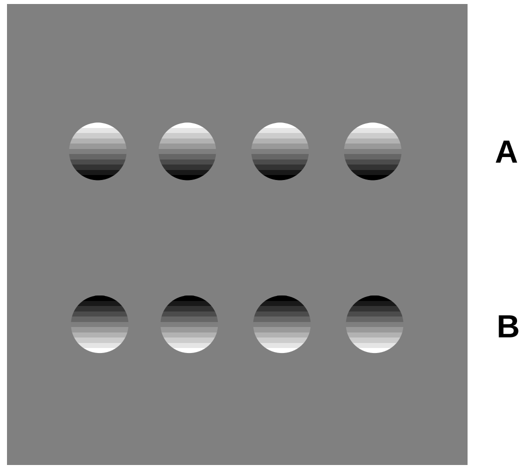


Figure 2-3: Demonstration of overhead lightsource

Due to the biological assumption of an overhead lightsource, row A is generally seen as convex and row B as concave.

& Sternfels, 1979; Wenderoth & Hickey, 1993; Howard, Berstrom & Ohmi, 1990). The convex-concave perception is strongest when the objects are shaded top to bottom relative to the observer's retina - not relative to the external world. If you bend over and look at Figure 2-3 through your legs with the figure oriented upright with respect to the world, but upside down with respect to your head, you will find that group A now appears concave and group B convex.

2.1.3 Other Cues Resolving Shape from Shading Ambiguity

Since the discovery of the ambiguity of shape from shading perception, people have been studying the various aspects of the phenomenon. As discussed previously the single and overhead lighting assumptions for resolving shape ambiguity have been much researched. Researchers have also studied cues other than shading which may help to resolve ambiguity in surface relief.

Figure 2-4: Is it a crater or a hill?
Occluding edges determine whether this image is perceived as a crater or a hill. If the occluding edge is at the bottom, it is perceived as a crater. If the occluding edge is at the top (hold the image upside down), it is perceived as a hill. Photo printed in Levine/Shefner, 1991 courtesy of U.S. Dept. of Energy.

Shape is influenced by binocular disparity in the positions of shading and shadows (Puerta et al, 1989), by texture gradients (Georgeson, 1979), by highlights on shiny surfaces (Todd & Mingolla, 1983), and by the distribution of luminance gradients over local regions (Pentland, 1982).

Occluding edges can also bias the way people resolve the depth ambiguity (Howard, 1983; Ramachandran, 1988b; Koenderink and van Doorn, 1982; Koenderink, 1984). An occluding edge delineates the boundary of a shape and thus makes explicit the fact that the shape may occlude other objects. For example, when viewing Figure 2-4, the occluding edge is at the bottom; therefore it looks like a crater (i.e., concave). When viewing this picture upside down, the occluding edge is at the top, so it looks like a hill (i.e., convex). Figure 2-5 exhibits another example of the influence of occluding edges. Both shapes were produced using the same shading pattern, although the left object looks like a row of cylinders, whereas the right object looks

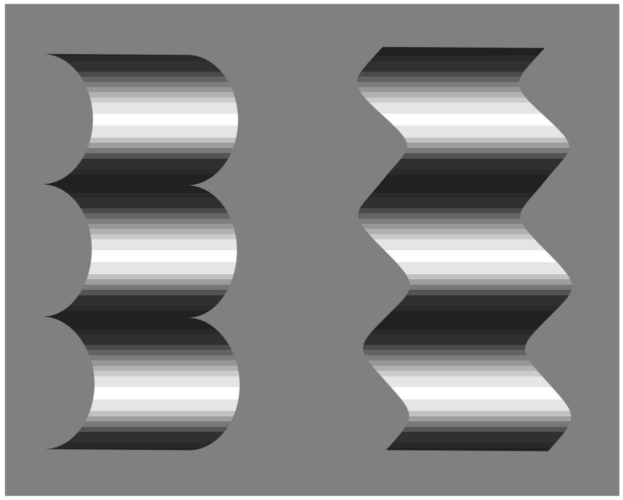


Figure 2-5: Edges affect the perception of shape
Both shapes were produced using the same shading pattern, although the left image looks like a row of cylinders, whereas the right image looks like a corrugated pattern.

like a corrugated pattern. These occluding edges are so convincing that they also appear to have different light sources, overriding the single light source assumption.

Furthermore, high-level knowledge of what is being seen profoundly affects our perception. The image in Figure 2-3 was described as convex objects in the top row and concave objects in the bottom row. However, this image was actually constructed by placing a flat grey rectangle with holes in it over a grating pattern. With this knowledge, the image can now be seen as blurred stripes viewed through holes in an opaque sheet.

Familiarity can also powerfully influence the perception of a three-dimensional shape. Objects that are familiar in a particular depth may bias the observer to see the object in the typical depth regardless of the true depth. For instance, faces are seen only as convex shapes. Thus, when seeing an image of a hollow, concave mask, there is a strong tendency to see it as a convex face, changing the direction of the light source to become consistent with the view of the face (Schroder, 1858; Gregory, 1970). In Figure 2-6 is bust of a male’s head on the left and a hollow replica or mask of a face on the right both lit from above. Both appear as a normal faces, even though the direction of light appears to be reversed (from below) for the hollow mask. The probability of hollow faces is so low that it is almost impossible to see the hollow mask correctly without some other cues to resolve the ambiguity.

Figure 2-6: Familiarity affects shape perception.
The exclusive familiarity with the convexity of a face prevents us from seeing the object on the right as a hollow mask. Photo printed in Gregory/Gombrich, 1973 courtesy of Gerald Duckman & Company, Ltd.

2.1.4 Discovery of the Effortless Perception of Depth From Shading

In a study to design an ideal 3D CAD system, Pentland (1987) found that concave spherical targets among convex distractors are processed in parallel, that is discriminations are made ‘instantly’ and involuntarily. Pentland argued that if the basic elements, blobs and deformations, are available preattentively and in parallel, then he could argue that this representation is one that closely matches the human ‘gestalt’ of the overall shape. Such a representation will be especially useful in the early stages of design, where the designer is attempting to ‘rough in’ the overall form and wants rapid control and feedback concerning the global characteristics of the shape.

Shape from shading perception involves primitive, low-level visual processing as is evident from the retina-centric assumption of an overhead light source and from the countershading found in the animal kingdom even though the perception of shape from shading is affected by other factors. Early, low-level visual processing is often associated with preattentive processing. Therefore, Pentland’s (1987) discovery that concave targets among convex distractors are processed in parallel is not surprising. Furthermore, the instantaneous, involuntary processing of depth from shading makes it a good candidate for an attentional cue in user interfaces.

2.2 Empirical Research Theory

The perception of depth through shape from shading cues is often studied by investigating the parallel versus serial distinction of the subjects’ performance. To understand this performance distinction, theories of attention must be considered.

2.2.1 Attention

Modern theories of attention started with Broadbent in 1957. He proposed that there is a limited capacity filter between the sensory channels and attentive stages of input analysis. A pre-categorical analysis is performed such that certain physical features are discernible (i.e., pitch and size) while semantic features are not (i.e., content, context, or meaning). This pre-categorical analysis allows only one message at a time to pass from sensory memory to the attention system. Triesman (1985) and others considerably modified Broadbent’s model to incorporate the fact that the stimuli in unattended channels do reach active attention. While these stimuli are severely attenuated, they have not ceased to exist.

Broadbent's theory consists of a two stage attentive system, where some input can be processed in parallel; whereas others require further individual processing, where each item is processed in turn. This type of dual system has received many names over the years. Neisser (1967) proposed a theory of attention based on a distinction between 'preattentive processes' and 'focal attention'. Similarly, Schneider and Shiffrin (1977) made the distinction between 'automatic processes' and 'serial-controlled processes'. In these models, focal attention and serial-controlled processes both describe a serial processing stage: discriminations can be made only when closely attended to. That is, only one object can be attended to at any given moment and each attentive act takes time. These models also claim that the preattentive and automatic processes, operating in parallel, use the simultaneously-arriving multiple stimulus information to construct the separate sensations involved. Discriminations occur 'instantly' and involuntarily. Neisser claims through his model that the preattentive stage of processing results in crude impressions of the stimuli's properties (e.g., movement, general location, brightness), whereas Shiffrin and Schneider claim through their model that the automatic stage of processing could result in meaningful content and context if the multiple inputs were well learned.

More recently, Kahneman (1973) and others have proposed capacity models advancing the idea that attention consists of a group of cognitive processes that can be allocated to deal with incoming information. The processing resources for any system are limited, and when several processes compete for the same resources, there will be an eventual deterioration of performance. Thus, when a user has a particular task to do using a computer, a certain amount of cognitive resources are allocated to this task. And when the user interface also requires additional cognitive resources to be processed properly, increased the cognitive load of the user potentially results in a deterioration of the user's performance. Further, demanding tasks require more resources than less demanding tasks, but this is true only for unpracticed demanding tasks. With practice, the mental effort required to do demanding tasks decreases, and if practice is continued, the processing of a task may become automatic, requiring virtually no resources and leaving no conscious trace of its execution. This is assuming that during the practice, the mapping between the stimulus and the response is consistent since a varied mapping usually prevents automaticy. Displays that can be processed preattentively or automatically do not require additional attention, leaving the user's entire capacity for the task at hand.

2.2.2 Parallel versus Serial Processing and the Visual Search Task

One paradigm that is commonly used to determine if particular items are processed in serial or in parallel is the visual search task. In a visual search task the subject is presented with multiple items simultaneously, among which there may be a target item. The subject responds according to whether the target item was seen. The slopes of reaction time per item displayed for the target-present trials and the target-absent trials are analyzed separately to determine how much additional time is required by each item in the display. The slope of reaction time per item displayed is determined by varying the number of items displayed from trial to trial. For example, consider the display in Figure 2-7. Several studies have shown that this type of display is processed in parallel, that is, the subject is able to determine if there is a concave object (the target) among the convex objects (the distractors) in an amount of time that is independent of the total number of items. However, compare this result with that of the display in Figure 2-8. This type of display is processed serially; that is, the subject's response time will increase proportionally to the total number of items displayed. Traditionally, slopes of 35 milliseconds per item for target-present trials and 70 milliseconds per item for target-absent trials have been interpreted as serial processing. The slope of the target-present trials being half of the slope of the target-absent trials may be due to a self-terminating search strategy. When the target is absent, the subject must inspect each object before determining that indeed the target is absent. However, when the target is present, the subject will find the target, on average, half way through the inspec-

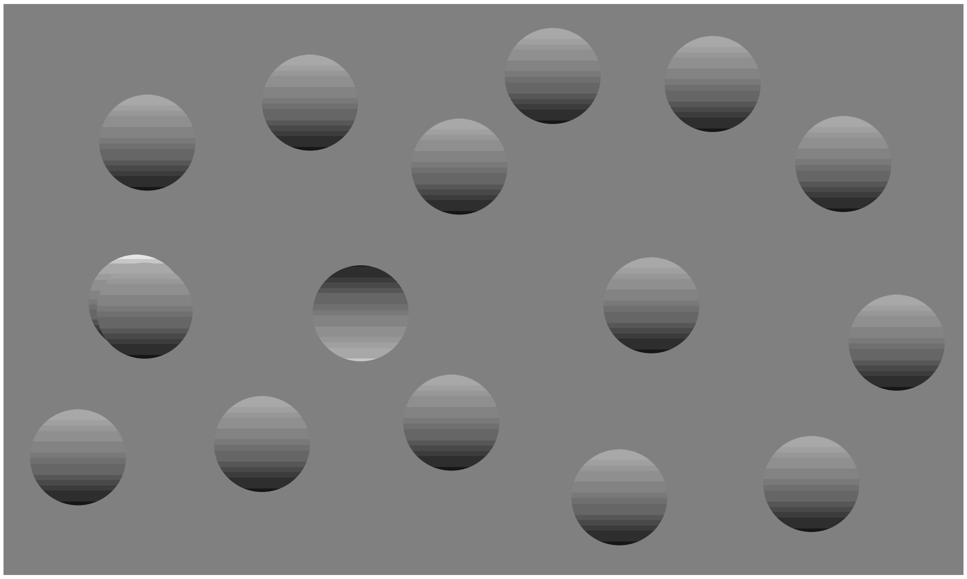


Figure 2-7: Display that can be processed in parallel

The subject can find the target (the concave object) in a roughly constant amount of time regardless of the number of total objects.

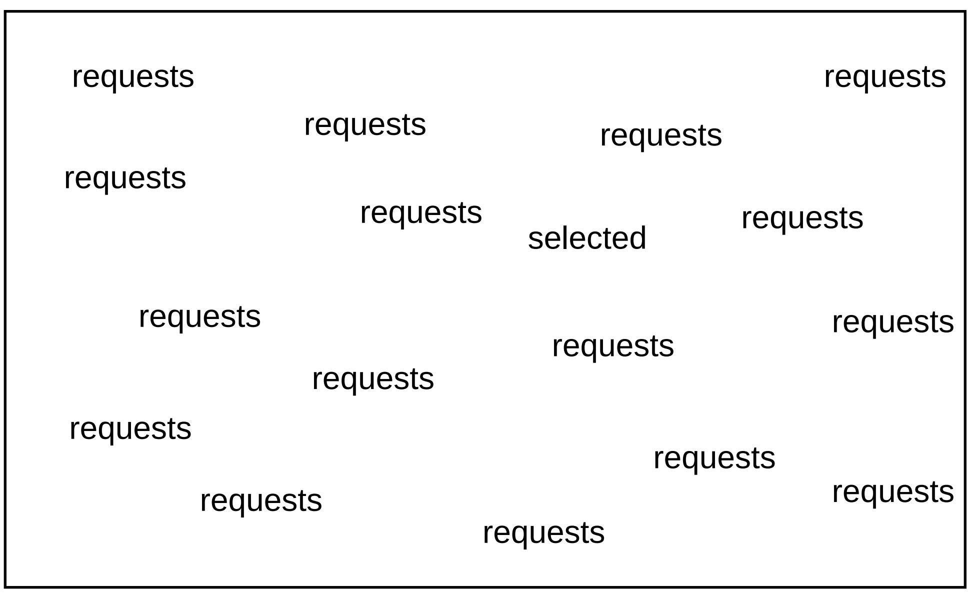


Figure 2-8: Display that is processed serially.

The subject's reaction time for finding the target increases linearly with the number of total items displayed.

tion. Slopes less than ten milliseconds per item for both target-present and target-absent trials have been interpreted as parallel processing. However, these slopes should not be interpreted as strictly enforced boundaries. The interpretation of the range of slopes between ten milliseconds per item and approximately 25 milliseconds per item is less clear. It is not necessarily reasonable to claim that a slope of ten milliseconds per item indicates parallel processing, whereas a slope of eleven milliseconds per item indicates serial processing.

2.2.3 Early Vision Processing

It is widely thought that discriminations that can be done in parallel have ‘hard-wired’, specific, ‘feature detector’' associated with them and are accomplished early in the visual system. After all, one cannot search for information that is not extracted from the input in the first place. Conversely, serial searches require attention and thus must be done much later in the visual system. Furthermore, perceptions made early in the visual system are likely associated with parallel processing. This is consistent with Broadbent’s (1957) theory of the attention system: the limited capacity filter initially processes the input in parallel. It is only then that individual sensory messages are passed to the attention system, one message at a time.

Symmetry experiments suggest that perception of depth through shape from shading is an early vision operation (Kleffner & Ramachandran, 1992; Howard et al, 1990, Wenderoth & Hickey, 1993). In symmetry experiments, the subjects are presented with a display such as in Figure 2-9. They are to respond quickly and accurately as to whether the display is symmetrical about the dotted line. Generally, this display is seen as symmetrical. However, when viewing a display such as this same figure viewed rotated 90°, this display is not seen as symmetrical. This means that when the objects are consistent with the overhead, retina-centric assumption, the three-dimensional shapes form the basis of the symmetry decision. However, when lighted from the side, the gradation patterns form the basis of the symmetry decision. So it appears

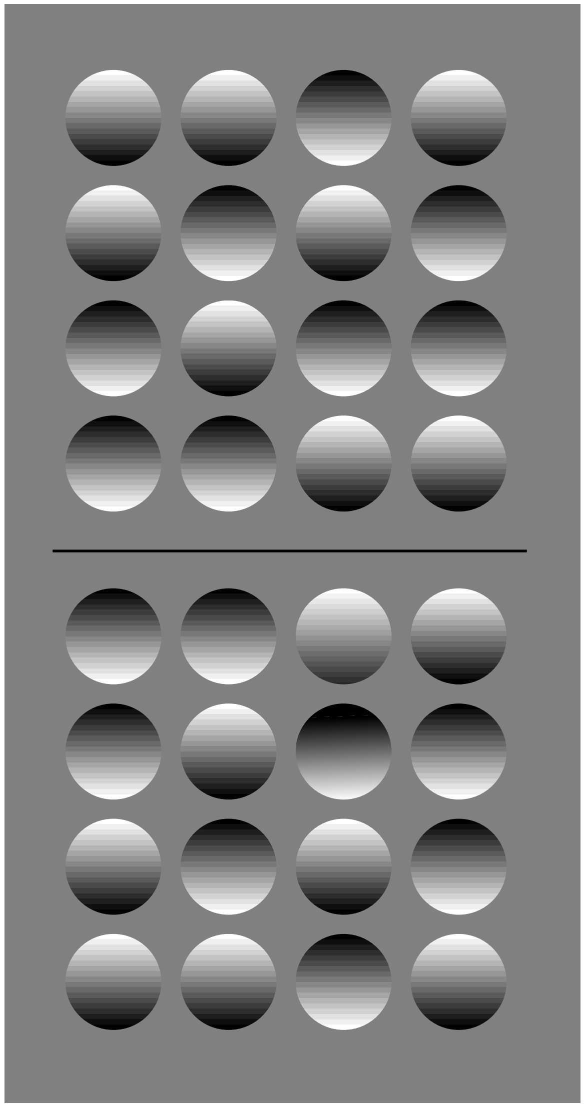


Figure 2-9: Is this display symmetrical?
With the overhead lighting assumption satisfied, this display appears symmetrical. However, it the overhead lighting assumption is not satisfied (hold this image at 90˚), this display does not appear symmetrical.

that shape from shading perception, given the overhead lighting assumption, happens early in the visual process and, thus, is processed in parallel.

2.3 Empirical Studies of Shape from Shading

There has been much support for the statement that shape from shading objects can be processed in parallel (Pentland, 1987; Kleffner & Ramachandran, 1992; Enns & Rensink, 1990; Sun & Perona, 1993; Braun, 1993) despite the use of different research methods. Pentland (1987), Enns and Rensink (1990) and Kleffner and Ramachandran (1992) measured the reaction time for visual search in static displays. Sun and Perona (1993) measured the percentage correct for visual search in transient, masked displays. Braun (1993) measured the percentage correct for two concurrent tasks in transient, masked displays. For the most part, the egg and egg-crate type shapes, as in Figure 2-7, are used to study this phenomena, most likely because they produce the strongest effects. Even so, Enns and Rensink came to the same conclusion using cubic shapes. Symmetry experiments have also been conducted which suggest that shape from shading perception is an early vision operation (Kleffner & Ramachandran, 1992; Howard et al, 1990). Recall that perceptions that occur early in the visual system are often processed in parallel.

One major finding of the Kleffner and Ramachandran studies was that the subject's ability to detect the target shaded vertically (i.e., consistent with an overhead light source) is significantly different from the subject's ability to detect either horizontal shading (i.e., consistent with a light source from the side) or a step change in luminance polarity. A step change in luminance polarity (illustrated in Figure 2-10) addresses the concern that it is actually the change in luminance that is the critical factor rather than the shading. They reported a four millisecond slope for the target present condition and a five millisecond slope for the target absent condition for vertical shading, a 22 millisecond slope for target present and a 50 millisecond slope for

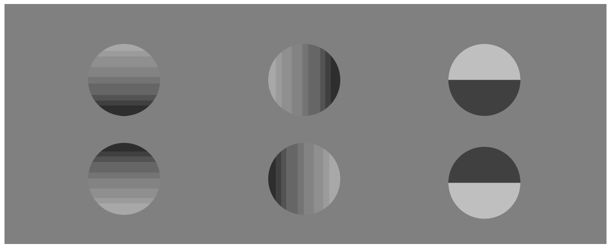


Figure 2-10: Objects in Kleffner and Ramachandran's (1992) experiment

target absent conditions for horizontal shading, and a eight millisecond slope for target present and a 18 millisecond slope for target absent conditions for luminance polarity. A three-way ANOVA analyzing all three conditions at the same time as well as pair-wise ANOVAs indicates that horizontal shading is significantly different than both vertical shading and luminance polarity. Thus, it seems that the extraction of shape from shading can provide a basis for effortless 'preattentive' visual search and that this is based on shading, not on luminance polarity, with the assumption that the light is shining from above. Kleffner and Ramachandran also found that naive subjects find concave objects easier to detect than convex objects, however, this is not true of experienced subjects.

Braun (1993) replicated Kleffner and Ramachandran's finding that concave objects are easier to detect amongst convex objects than convex amongst concave objects. However, Braun measured percentage correct where the stimulus is transient and masked, whereas Kleffner and Ramachandran measured reaction time where the stimulus is static. Braun also found that shape from shading perception is processed in parallel using a concurrent task paradigm. In a concurrent task experiment, one task is used to draw limited resources away from the other task, demonstrating that a particular task does not draw on a particular resource. In this study, one concave target could appear in the top half of the display, and independently, another could appear in the bottom half of the display. The stimulus was investigated under two single-task and one double-task condition. The study found that separate and concurrent detection produced essentially identical performance, which provides additional evidence that shape from shading is processed in parallel. Further, Braun asked whether shape from shading perception demands visual attention. To this end, he conducted an experiment similar to the previous concurrent task, however, one task was a presence/absence determination of the shaded object and the other task was a T/L discrimination task (the subject indicated whether the letter was a T or an L). Similar to the previous concurrent task experiment, the stimulus was investigated under two single-task and one double-task conditions resulting in essentially identical performance. Therefore, given that the T/L discrimination places a significant demand on visual attention (determined to be 60ms), the observed outcome implies that detecting a shape from shading target does not.

Unlike the stimuli of the previous studies, the basic component in a windowing interface that uses shape from shading cues is likely to be block or button shapes. Moreover, these block shapes will be composed of three intensities (one for each visible face), rather than smoothly varying intensities, as in the egg and egg-crate stimuli. Enns and Rensink (1990) reported that search for shaded polygons is rapid when these items can be interpreted as three-dimensional blocks. However, rather than studying convex versus concave blocks under a single lightsource, they studied convex blocks with varied lighting directions: they compared lighting from above and lighting from the side. They found that this preattentive feature seems to be the deviation from the standard direction of lighting assumption, that is, lighting from above. Further, they found that these effects did not require intensities to be varied smoothly - three intensities were sufficient.

2.4 Validity of Serial versus Parallel Analyses

Some researchers think that the linear time effects of serial processing and constant time effects of parallel processing are too naive. By rapid switching, serial systems may mimic parallel ones, and parallel systems of limited capacity can show linear time effects. Consider the analogy of an elevator with a weak motor. As more people get on the elevator, the elevator rises more slowly, yet the people are still moving in parallel. Also, poor strategy choices by the subjects can result in the slope of the target-absent trials to be double of the slopes of the target-present trials. For instance, if the target is processed in parallel and the subjects respond quickly, they may not trust their perception that the target is absent and thus will actively search for the target, thereby increasing the reaction time of the absent trials. There are ways to reliably assess whether the operation is parallel or serial (Townsend, 1990). However, it is not necessarily of interest whether the perception of shape from shading as a windowing system cue is parallel or serial. It is more interesting and more practical to ask whether it is fast for most reasonable windowing displays. Thus, slopes less than ten milliseconds per item will be considered as meeting our criterion of being fast. This criterion of ten milliseconds per item was chosen since it is one of the distinguishing criteria between parallel and serial processing. The second criterion (the slope of target-absent trials being double the slope of target-present trials) is considered unimportant since it does not directly address the question of speed of processing. Furthermore, this criterion simplification certainly does not invalidate the previous research on shape from shading processing.

2.5 Statistical Analyses

To determine if the reaction time per item displayed is less than ten milliseconds a linear regression analysis is used followed by a 95% confidence interval analysis on the slope. Regressions are used when the specification of a relationship is desired, such as the relationship between the number of items displayed and the subject's reaction time: the experimenter is interested in showing that the dependent variable is some function of the independent variable. The value of the independent variable is fixed or specified by the experimenter before the data are collected. Thus, no sampling error is involved in the independent variable, and replications of the experiment will involve the same independent variable values.

A linear regression determines the best-fitting line for predicting Y (dependent variable) on the basis of X (independent variable): $\hat{Y} = bX + a$. Given that $\hat{Y}$ is the predicted value and Y is the actual value, $(\hat{Y} - Y)$ is the error of prediction. The linear regression is the line that minimizes such errors. Since any line that goes through a point $(\bar{X}, \bar{Y})$ will always result in the sum of the errors being equal to zero, simply minimizing the sum of the erros is not sufficient. Thus, the sum of the squared errors should be minimized.

To obtain the optimal values of a and b - that is, to minimize the errors of predition in the linear regression - the equation

$$\sum (Y - \hat{Y})^2 = \sum [Y - (bX + a)]^2$$

is solved resulting in $b = \frac{cov_{XY}}{S_X^2}$ (covariance of X and Y divided by the variance of X) and $a = \bar{Y} - b\bar{X}$, where $cov_{XY} = \frac{\sum (X - \bar{X})(Y - \bar{Y})}{N - 1}$ and $S_x^2 = \frac{\sum (X - \bar{X})^2}{N - 1}$ (Howell, 1992).

The confidence interval on the slope, b, is determined by $CI(b) = b \pm (s_b \times t)$, where t is the tabled t-ratio with a two-tailed probability of 0.05 and the required degrees of freedom. From this analysis, we can be 95% confident that the actual slope lies within this interval. Thus, if the upper bound of this interval is less than the required ten milliseconds then the slope is less than ten milliseconds (p = 0.05).

In controlled, fixed experiments, ANOVAs and regressions are proper tests for testing the differences between or among sample means and for predicting the relationships amongst variables, respectively. However, in an experiment with random independent variables, replications of the experiment will not involve the same independent variable values and thus sampling error is involved in both the dependent (Y) and the independent (X) variables. Therefore, correlational tests are performed to measure the degree of relationship between these two variables.

The Person Product-Moment Correlation Coefficient (r) is defined as the covariance divided by the standard deviations of the variables: $r = \frac{cov_{XY}}{S_X S_Y}$, where cov_{XY} is defined as above, and $S_X = \sqrt{S_X{}^2} = \sqrt{\frac{\sum (X - \bar{X})^2}{N - 1}}$ and similarly $S_Y = \sqrt{\frac{\sum (Y - \bar{Y})^2}{N - 1}}$ (Howell, 1992).

A test of the significance of a correlation tests whether variables X and Y are linearly independent. It can be shown that when the correlation of the actual population is zero, for large N, r will be approximately normally distributed around zero with a standard error that can be estimated by $s_r = \sqrt{\frac{1 - r^2}{N - 2}}$

Since the distribution of r is approximately normal and its standard error is estimable, the t-ratio can be determined as $t = \frac{r}{s_r}$, which is distributed as t on N-2 degrees of freedom. The significance of this t-ratio can be determined in comparison to the tabled t-ratio at alpha = 0.05 and the required degrees of freedom for a two-tailed test.

When analyzing the correlations among several independent variables and the dependent variable, the correlations within the independent variables can help to explain the data. For instance, consider the case where variables A and B are both positively correlated with the dependent variable, Y, and A and B are positively correlated. It is possible that A is truly correlated with Y and B is correlated with Y only due to its correlation with A, or vice versa. Also consider the case where variable A is positively correlated with Y and B is negatively correlated with Y. Furthermore, A and B are positively correlated. In this case, it is likely that one of the correlations is spurious. To illustrate, as B increases, A increases. As A increases, Y increases. But as Y increases, B decreases. These relations cannot all be satisfied simultaneously, therefore, at least one relation is likely to be spurious.

2.6 Summary

It is apparent that when designing an interface using depth from shading as an attentional cue, the apparent light source should be overhead and there should be only one apparent light source. Moreover, the direction of the light source should be identifiable. Also, to study empirically whether depth from shading cues can be used to indicate the active window effectively, visual search tasks are appropriate. Slopes of ten milliseconds per item can be used to interpret the effortlessness of the processing.

Chapter 3

Experiment Implementation

Many vision experiments have similar components and must deal with the same issues. A typical vision experiment has several trials which may or may not be presented in blocks. Each trial may consist of several components. For example, a typical trial may be presented as follows. First, a fixation point appears in the centre of the display area and remains on the screen for a specified period of time. Next, the fixation point is replaced with the stimulus. The stimulus image may remain on the screen until the user responds or may remain on the screen only for a specified stimulus onset asynchrony (SOA) duration. If the image is presented for a specific SOA duration, then typically the image is replaced with a mask to prevent further processing of the image in the subject's iconic visual memory. This mask is typically presented for a fixed period of time and is then replaced with a blank screen. Usually both the reaction time and the accuracy of the response is measured. Accurate timing is of crucial importance to this type of experiment as is the instantaneous appearance of the stimuli. There would be a serious confound if parts of the stimulus image consistently appeared before the rest of the stimulus image.

When implementing the experiments for this research, it appeared beneficial to create an experimental framework where the experimental structure and timing issues could already be resolved. The experimental framework, implemented in C++, is a forest data structure (Reingold & Hansen, 1945): the application and experimental control form a tree and other trees are present that provide help to a programmer to create a new experiment (Figure 3-1). The application class controls events related to the application before the actual experiment starts. The experimental application class controls events related to the experiments. This class uses other support classes to handle particular issues, such as timing and parsing the experiment information from a file.[1]

3.1 Application Control

The application class (CApp) controls and handles the basic application features, such as providing support for creating a menubar, handling the menu events, mouse events, and key events. This class uses an event loop to get and process the events.

3.2 Experimental Control

The experimental application class (CExptApp), derived from CApp, maintains control of the experiment. It defines the menus that are supported by CApp. There are two default menus: a File menu, which has only a Quit option, and an Experiment menu, which has Filenames, Subject ID, and Run options. The Filenames menu option presents a dialog box which prompts the user to specify the filenames of the input and output files. The Subject ID menu option prompts the user to enter the alphanumeric ID of the subject. The Run option starts the experiment.

---

1. All classes were developed (i.e., they were not provided by a commercial software company). The CApp and CLex classes were developed by Greg Veres (member of CGL). CMenuBar was developed by UofT (as C code). These classes were modified to suit the experimental framework. All other classes were written by the author of this thesis.

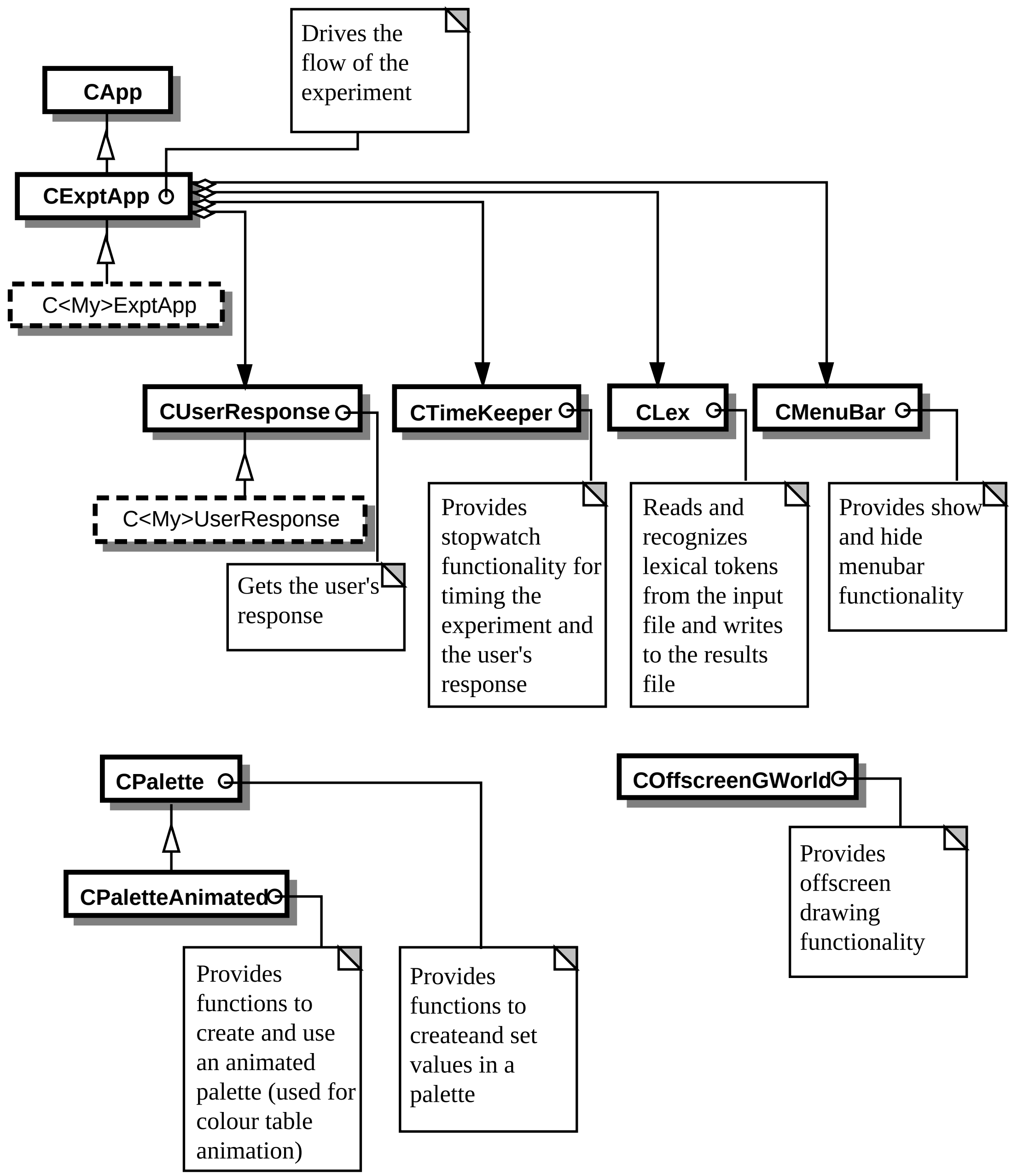


Figure 3-1: Experiment Framework

The class CExptApp drives the experiment and the classes CUserResponse, CTimeKeeper, CLex, and CMenuBar are instantiated as data members in CExptApp. The classes CPalette, CPaletteAnimated, and COffscreenGWorld are provided to the programmer to aid in displaying the images.

The code that controls the execution of the experiment is presented in Figure 3-2. The code consists of a preliminary setup procedure where things can be done once prior to the execution of the first trial, followed by a loop which is executed once per trial. To create an actual experiment from this experimental framework, a derived class needs to be created from the CExptApp class. This allows the programmer to define functionality within the predefined procedure calls shown in the code of Figure 3-2. Many of the procedures have default actions and thus for most cases some of these procedures do not have to be redefined.

3.2.1 Preparatory Setup

Initially, input and output files are opened (`this->OpenFiles()`). The file names are specified by the experimenter through an option in the menubar (however there are default file names). If the files are opened successfully, the preliminary setup phase is executed (`this->PrepareExperiment()`), which creates a canvas display window (`this->SetupWindow()`). The display window is presented such that the canvas area of the window occupies the whole monitor's display and none of the window borders or titlebar are visible. This window then becomes the canvas area for the presentation of the experimental trials. After the window is created, notes to the experimenter are presented in a dialog box (`this->ShowNotesToExperimenter()`). Primarily, they remind the experimenter to disconnect the Macintosh from the network, because processing of the network activity requires CPU time which may affect the experiment timing. Thus, the computer should be disconnected from the network before proceeding with the experiment. Next, the cursor can be setup (`this->SetupCursor()`). If the preprocessing of the images, which occurs later, is a lengthy process, then feedback in the form of an animated cursor may be desired. The cursors are created or read in from a resource. Note however, that there is no default behaviour for setting up the cursor.[2]

The experimenter can then type in comments regarding this particular experiment (`this->ShowCommentsDialog()`), which are copied to the output file. The experiment programmer may then read in and deal with information from the input file before proceeding further with the experiment (`this->ReadInitialInput()`). In the case of the first two shape from shading visual search experiments, the information regarding the direction of the light source, the definition of the display palette, the colour of the background, and the definition of the shapes to be used in the experiment were read in and dealt with accordingly.

Next, the programmer may then create additional palettes or an animating palette to be used for colour table animation (`this->SetupPalettes()`). Colour table animation is discussed below (Section 3.4). Any necessary preprocessing of the images, such as drawing offscreen images which can be copied to the display area later in the experiment (`this->PreprocessImages()`). Note that there is no default behaviour for the ReadInitialInput, SetupPalettes, and PreprocessImages procedures. Next, the cursor is hidden (`this->HideCursor()`). If the cursor is needed in the experiment, the programmer can override this procedure with an empty procedure. Finally, the programmer presents information about the experiment to the subject (`this->ShowTaskDescriptionDialog()`). For example, the dialog may state that the following is a visual task experiment and the subject is required to hit the 'm' key if the target is present and the 'c' key is the target is absent. This procedure has no default behaviour.

3.2.2 Trial Loop

The loop which presents the trials executes after the preliminary setup is complete. The processing of the loop proceeds until there are no more trials to present or until the subject requests an end to the experiment. Note that the procedures in this loop provide only the structure of the experiment; the functionality must be provided by the pro-

2. When it is stated that a procedure has no default behaviour, it is not intended to mean that it is a pure virtual function which must be overridden, but rather that the default behaviour has no functionality - it is a no-op(eration).

```
// -------------------------------------------------------------
//    • ExptRun
// -------------------------------------------------------------
void CExptApp::ExptRun()
{
  if ( this->OpenFiles() ) {

     this->PrepareExperiment();

     Boolean quit = FALSE;
     while ( this->NextTrial() && !quit ) {
        quit = this->ExecuteTrial();
     }

     this->CleanUp();
  }
}

// -------------------------------------------------------------
//    • PrepareExperiment
// -------------------------------------------------------------
void CExptApp::PrepareExperiment()
{
  this->SetupWindow();

  // show notes to the experimenter, such as
  // 'Remember to disconnect the computer from the network'.
  this->ShowNotesToExperimenter();

  // get cursors to be used as animating cursors
  this->SetupCursor();

  // display a dialog where the experimenter can type comments
  // which will then be copied to the output file
  this->ShowCommentsDialog();

  this->ReadInitialInput();

  // create additional palettes or an animating palette to be
  // used for colour table animation
  this->SetupPalettes();

  // perform any preprocessing (eg. drawing images offscreen)
  this->PreprocessImages();

  this->HideCursor();

  // present description of experimental task to the subject
  this->ShowTaskDescriptionDialog();
}
```

Figure 3-2: Code for controlling the experiment

```
// ---------------------------------------------------------------
//    • ExecuteTrial
// ---------------------------------------------------------------
Boolean CExptApp::ExecuteTrial()
{
  // start the clock to time the inter-trial interval
  mTimeKeeper.Start();
  this->DrawFixationCross();

  this->PrepareImage();

  // delay for timeDelay milliseconds before starting the clock
  // and displaying the image
  mTimeKeeper.DelayUntil( mTimeDelay );
  // start the timer to measure the subject's RT
  mTimeKeeper.Start();

  mUserResponse->Flush();

  // Note: generally there are two ways of creating and displaying
  // the image:
  //  1)  the image is drawn directly to the screen and
  //      colour lookup table animation is used, OR
  //  2)  the image is drawn offscreen and then copyBit'ed
  //      to the screen
  this->DisplayImage();

  // if the image is to be displayed for a fixed amount of time,
  //     use the following routines
  // if the image is to be displayed until the user responds,
  //     Get*DisplayDuration should return 0 (which is the
  //     default) and DisplayMask and RemoveMaskAndDisplay
  //     should do nothing (which is also the default)
  mTimeKeeper.DelayUntil( this->GetImageDisplayDuration() );
  this->DisplayMask();
  mTimeKeeper.DelayUntil( this->GetMaskDisplayDuration() );
  this->RemoveMaskAndDisplay();

  // wait for the user's response and stop the clock
  quit = (mUserResponse->GetUserResponse() == UserResponse_QUIT);
  mTimeKeeper.Stop();

  // if not already removed (above), remove the image now
  this->RemoveImage();

  // write the information to the file
  unsigned long RT = mTimeKeeper.GetElapsedTime();
  this->WriteTrialInfo( RT );
}
```

Figure 3-2: Code for controlling the experiment (Continued)

grammer.

The trial starts by drawing the fixation point. While the fixation point is being displayed, the stimulus image is prepared. The fixation point remains on the screen until a specified amount of time (`mTimeDelay`), measured in milliseconds, has passed; the default duration is 1000ms. Next, the user response is flushed so that no events remain in the queue prior to displaying the stimulus. Then the image is displayed. If the experimental paradigm is to present the stimulus for a fixed SOA, then to be replaced by a mask which remains on the screen for a fixed duration, then the GetImageDisplayDuration, DisplayMask, GetMaskDisplayDuration, and RemoveMaskAndDisplay procedures should be overridden with the appropriate functionality. If the stimulus is to remain on the screen until the user responds, then the preceding procedures are not needed and the default functionality works appropriately in this case. That is, GetImageDisplayDuration and GetMask-DisplayDuration both return 0 and DisplayMask and RemoveMaskAndDisplay both do nothing. Next, the user's response can be acquired and the timer is stopped. The image is then removed through the RemoveImage procedure. Finally, the elapsed reaction time is determined and the information is written to the output file.

As indicated in the procedure NextTrial, some basic behaviour is provided in response to particular tokens in the input file (see code in Figure 3-3). Only the MoreTrialInfo procedure must be overridden to read in information appropriate for the new experiment. NextTrial will recognize the tokens EOF, BREAK, END_DESCRIPTION, START_PRACTICE, END_PRACTICE, START_FIXATION_FEEDBACK, END_FIXATION-_FEEDBACK, and ACCURACY_FEEDBACK. It is expected that break will be used to provide a break for the subjects between blocked conditions. BREAK displays a dialog box which states that the subject should press <return> to continue. A description of the next block of trials can be presented in the Break dialog box by providing the description in the input file. Following break on the same or next line should be the comment symbol (#) and the description, where the whole description is on one line. The next line must have the END_DESCRIPTION token.

START_FIXATION_FEEDBACK and END_FIXATION_FEEDBACK sets a flag indicating whether feedback is needed by way of the fixation point. If fixation feedback is desired, the DrawFixationCross procedure should be coded such that different fixation images are drawn depending on the state of the fixation flag. In all three of the experiments fixation feedback was presented: if the response to the previous trial was correct then the fixation point was a ‘plus’ sign and if the response was incorrect then it was a ‘minus’ sign. The START_PRACTICE and END_PRACTICE tokens delimit the practice tri-

```
// ----------------------------------------------------------------
//    • NextTrial
// ----------------------------------------------------------------
//  Reads the next trial information. Returns TRUE unless eof
//  reached.

   // indices into cToken to get the desired token
   const int cEOF       = 0;
   const int cBREAK  = 1;
   const int cSTART_PRACTICE= 2;
   const int cEND_PRACTICE= 3;
   const int cEND_DESCRIPTION= 4;
   const int cSTART_FIXATION_FEEDBACK= 5;
   const int cEND_FIXATION_FEEDBACK= 6;
   const int cACCURACY_FEEDBACK= 7;

   // number of tokens in the cToken array
   const int cNumTokens= 8;

   static const CToken cTokens[cNumTokens] = {
      { tEOF,                        cEOF                        },
      { tBREAK,                      cBREAK                      },
      { tSTART_PRACTICE,             cSTART_PRACTICE             },
      { tEND_PRACTICE,               cEND_PRACTICE               },
      { tEND_DESCRIPTION,            cEND_DESCRIPTION            },
      { tSTART_FIXATION_FEEDBACK,    cSTART_FIXATION_FEEDBACK    },
      { tEND_FIXATION_FEEDBACK,      cEND_FIXATION_FEEDBACK      },
      { tACCURACY_FEEDBACK,          cACCURACY_FEEDBACK          }
   };
```

Figure 3-3: Code for reading next trial information from the input file

```
Boolean CExptApp::NextTrial()
{
  Boolean doneCurrTrial = FALSE, isMoreTrials = TRUE;
  do {
     switch ( mLex.MatchToken( cTokens, cNumTokens ) ) {
       case cEOF:
        doneCurrTrial = TRUE;
        isMoreTrials = FALSE;
        break;

       case cBREAK:
        if ( mLex.MatchToken( cTokens, cNumTokens ) ==
                                      cEND_DESCRIPTION ) {
           this->ShowBreakDialog( (char *)mLex.GetComment() );
        } else {
           this->ShowBreakDialog( "unknown" );
        }
        break;

       case cACCURACY_FEEDBACK:
        this->ShowAccuracyFeedback();
        break;

       case cSTART_PRACTICE:
        mIsPractice = TRUE;
        break;
       case cEND_PRACTICE:
        mIsPractice = FALSE;
        break;

       case cSTART_FIXATION_FEEDBACK:
        mIsFixationFeedback = TRUE;
        break;
       case cEND_FIXATION_FEEDBACK:
        mIsFixationFeedback = FALSE;
        break;

       default:
        doneCurrTrial = this->MoreTrialInfo( arg );
        break;
     }
  } while ( !doneCurrTrial );

  return ( isMoreTrials );
}
```

Figure 3-3: Code for reading next trial information from the input file (Continued)

als in the input file. Practice trials are not recorded in the results file. If the experiment does not read input from a file, as in Experiment 3, then the NextTrial procedure should be overridden.

3.3 Timing

The class CTimeKeeper is used like a stopwatch and provides accurate timing in milliseconds. The stopwatch functionality is provided through the Start, Stop, GetElapsedTime, DelayFor, and DelayUntil procedures. Accurate timing is provided by the Macintosh Toolbox procedure Microseconds, which can be accessed by including the Universal header Timer.h.[3]

3.4 Colour Palettes and Colour Table Animation

Another issue similar to the need for accurate timing is the need for near instantaneous appearance of the display images. If parts of the stimulus image consistently appears before the rest of the stimulus image, the subject's response strategy could be affected, particularly in a visual search task. Colour table animation, which draws the image by changing indices in a colour table, is an excellent technique for displaying the image instantaneously (Foley, van Dam, Feiner & Hughes, 1990). When another colour table is swapped in, the colours of the image change according to the colours associated with the indices in the image. Thus, if the image is to appear following a blank (say, white) screen, the image can be drawn while an all white colour table is associated with the drawing port. Then, when it is time to display the image, the colour table

3. The Microseconds documentation did not state the actual granularity of the Microseconds timer. But through tests that I conducted, I am convinced it is much smaller than the experiments' required millisecond granularity.

appropriate for the image is swapped in and the image appears instantly. Furthermore, note that the Macintosh synchronizes the colour table animation with the screen update. Thus, the technique of colour table animation is trivial if the animation involves at most one complex screen and all others are uniform in colour. However, consider the case where a nontrivial image is followed by another nontrivial image, such as a stimulus image followed by a mask. The image must be drawn in a manner such that once the stimulus image is displayed the mask will appear by only swapping in the colour table associated with the mask. This can be achieved by drawing a very complex image, which is the composition of both the stimulus image and the mask, and by appropriate design of the two colour tables. In this complex composition of the images, each unique area when the two images are overlaid is drawn with its unique index. With proper structuring of the colour tables and proper choice of indices, the image will animate from one image to the other by swapping the colour tables. Consider the images labeled (i) and (ii) in Figure 3-4. They consist of three and four colours respectively colours including the background colour. When these two images are overlaid, there are twelve unique mappings from a given colour in the first image to another colour in the second image. And each unique mapping is drawn with a unique colour table index. Note that the maximum number of unique areas is the product of the number of unique areas in each of the original images. Thus, for this example both of the colour tables must have twelve entries. The colour table associated with one image, say the first image, repeats each colour in this image in turn n times, where n is the number of colours in the second image. And the colour table associated with the second images repeats all colours in this image m times, where m is the number of colours in the first image. See Figure 3-5 for an illustration of the colour tables and associated images.

The structure of the colour tables can be modified if it is guaranteed that certain parts of the two images do not overlap. This was the case for Experiments 1 and 2. The stimulus shapes never overlapped with the fixation point, so the colour table associated with the fixation point display maintained one entry for the fixation point colour and

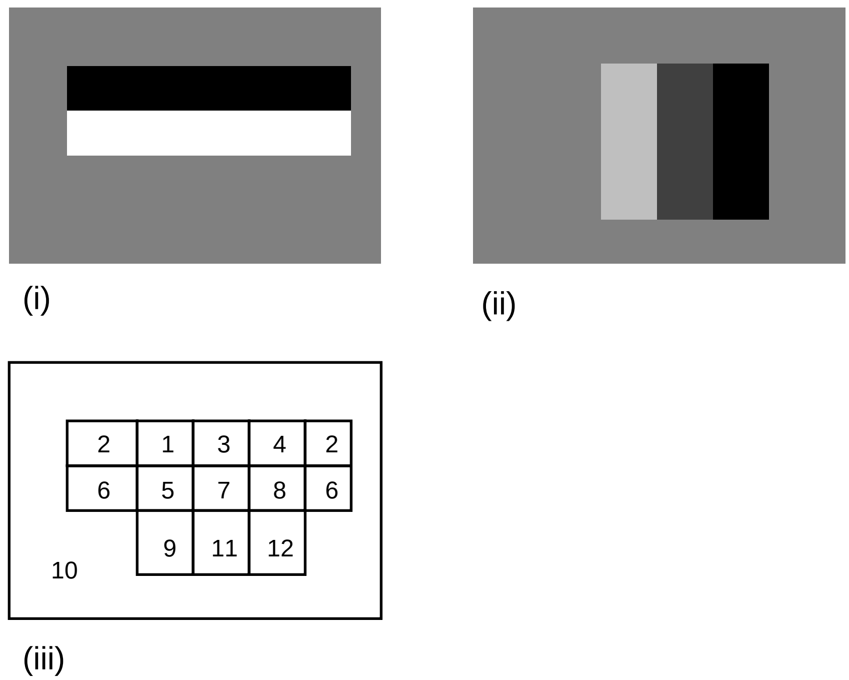


Figure 3-4: Colour table animation: create unique areas
To animate from image (i) to image (ii), areas are drawn such that mappings from a given colour in the first image to another colour in the second image are possible. Each unique mapping is drawn with a unique colour table index. Notice that in image (iii), there are two areas which are assigned colour table index 2 (and 6) since the mapping is the same.

the rest of the entries were the background colour. The colour table associated with the stimulus had the background colour in the entry associated with the fixation point in the other colour table and the rest of the entries were appropriate for the stimulus image. For Experiment 3, the fixation point and the stimulus objects may overlap. So it is necessary to use the blank screen method of colour table animation. The fixation point appears for the default duration. While the fixation point is being displayed, the stimulus image is written into an offscreen drawing area. At the end of the fixation point display duration, the blank palette is swapped in making the screen appear the uniform grey background colour. The stimulus image pixel map is then copied into the

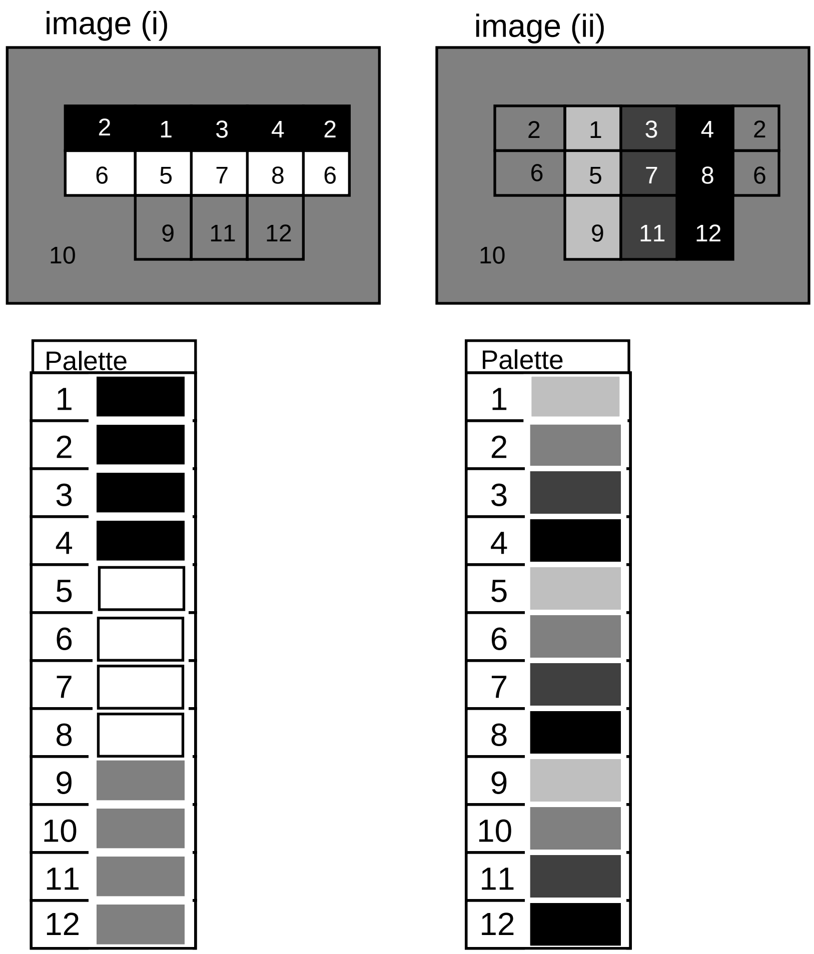


Figure 3-5: Colour table animation: animate using colour table

To show image (i), its associated colour table is swapped in. Then to show image (ii), its associated colour table is swapped in.

image memory and then the appropriate colour table is swapped in, resulting in the instantaneous appearance of the stimulus image.

Since colour table animation is instantaneous or near instaneous, timing issues are related to the amount of time it takes to create the images before the colour tables are animated. For all three experiments, the experimental images are guaranteed to be created in less than 1000 milliseconds, which is the default duration for which the fixation point is displayed.

Two classes, CPalette and CPaletteAnimated, support palette creation and colour table animation. CPalette provides CreateNewPalette, SetEntryColour, SetColourRange, GetColourTable and AttachToWindow, which allows easy colour palette creation and manipulation. CPaletteAnimated provides SwapColourTable and AttachToWindow functionality. Furthermore, CPalette provides an input procedure such that the palette information and thus palette creation can be initiated through the input file.

3.5 Validity of Experimental Framework

This framework provides structure for a wide range of visual experimental paradigms. For instance, the shape from shading visual search experiments presented a fixation point for a fixed duration. The fixation point was then replaced by the stimulus image which remained on the screen until the user responded. Thus, the only procedures which needed to be overridden within the trial loop were DrawFixationCross, PrepareImage, DisplayImage, RemoveImage, and WriteTrialInfo. In contrast, consider an experimental paradigm where a fixation point is presented for a fixed duration and is then replaced by the stimulus image for a fixed SOA duration. The image is then followed by a mask for a fixed duration upon which it is removed. In this case, the procedures which need to be overridden within the trial loop are DrawFixationCross, PrepareImage, DisplayImage, GetImageDisplayDuration, DisplayMask, GetMaskDisplayDuration, RemoveMaskAndDisplay, and WriteTrialInfo. Notice that accurate timings and acquisition of binary user responses are provided by default.

This experimental framework was also used to create a Rods and Frames experiment.[4] Once the implementation method was decided and the image bitmaps were created, only a couple of days was required to create a working experiment using this framework.

4. In a Rods and Frames experiment, a vertical rod is presented tilted with respect to the world. It is surrounded by a frame which may also be tilted, although the amount and direction of tilt are not consistent with one another. In the different trials, the rod is rendered using different anti-aliasing techniques. The subject responds indicating whether the rod was tilted or not. This paradigm is used to assess different methods of anti-aliasing.

Chapter 4

Experiments

A series of visual search experiments explored whether or not adding depth from shading cues to window designs would make the display more understandable, leaving the user's cognitive capacity free for his or her work. These experiments required subjects to respond to the stimuli indicating whether the target was seen. In the case of the experiments with window elements, most of the windows would be distractors and one window may be the target. Half the trials of an experimental session present a single target among several distractors (target-present trials); half of the trials present only distractors (target-absent trials). The distractors and target appear similar with the exception of the cue present in the target window that indicates that it is the target or active window (Figure 4-3). These trials parallel the way users must use multi-window interfaces: the user must recognize which window is active to be able to either use the active window or to make another window active for use.

4.1 General Methodology

The methodology of the following experiments consisted of presenting several trials of the stimulus on a Macintosh IIfx to which the subjects respond through the keyboard. A single trial consists of three components. First, a fixation point appears in the centre of the display area. Secondly, after a constant duration (one second), the fixation point is replaced with the stimulus image. The stimulus image remains on the screen until the subject produces a response, which is the third component.

For each trial, the subject directs his or her attention at the fixation point. Standardizing the direction of gaze reduces variability due to variations in the direction of gaze. If the subject's response was correct for the preceding trial, the fixation is a fixation cross or ‘plus’ sign and if the subject’s response was incorrect for the preceding trial, the fixation is a ‘minus’ sign. As was found in pilot studies, this feedback to the subject is essential for maintaining a low error rate.

In all experiments, there is a ‘number of elements’ condition and a ‘target presence’ condition. These are the basic conditions of a visual search experiment. That is, the stimulus is an arrangement of elements which varies from trial to trial. The target element may or may not be present and the remaining elements are considered the distractors. In the first two experiments, either four or sixteen elements are displayed in on 4×4 grid. For the four element condition, four random locations of the grid are chosen.

The random 4×4 grid placement arrangement was chosen over a variable-dimension grid, where sixteen elements are displayed on a 4×4 grid and four elements are displayed on a 2×2 grid, because the former arrangement would control better for foveal versus peripheral vision confounds. That is, all four elements on a 2×2 grid would be closer to the fixation point than would some of the elements displayed on a 4×4 grid.

The stimulus remains on the display until the subject indicates whether the target was present or not, which ends the trial. During the trial, the subject has a finger of his or her right hand poised to hit the ‘m’ key and a finger of the left hand poised to hit the ‘c’ key. The response is made by hitting the right-hand key if the target was present and the left-hand key is the target was absent. Both reaction time and accuracy is recorded. The subjects are instructed to maintain a high degree of accuracy because accuracy is a necessity in windowing environments. The dependent measure, reaction time, measures the speed at which the subjects can use the depth from shading cue to determine if the target is present. Consider the use of a windowing system. When the user attends to the relevant cues to determine the state, errors are almost never made. However, when errors do occur, it is when the user is not explicitly attending to the windowing state. Therefore, the subjects are instructed to fixate at the centrally located fixation cross prior to the stimuli and to respond as quickly as possible while maintaining a high degree of accuracy. This method is used since we do not want to monitor the behaviour when users or subjects are attending to cues; we want to monitor their behaviour when minimal attention is given to the task.

No counterbalancing of handedness with respect to the target response is necessary since, in the analysis, slopes are determined for each of the target-present and target-absent trials. Reaction time differences due to handedness are most likely contributing to intercept rather than slope effects.

The stimuli are displayed using a Macintosh IIfx on a DataTrain display screen. The stimuli are drawn using one or two pixel wide lines for the light highlights and the dark shadows on a medium grey field, where the flat plane of the objects are the same medium grey as the background field (Figure 4-3). The background field has a luminance of about 50 candelas per square metre. The subject views the screen from a comfortable viewing distance for workstation use, about 500 mm. To simulate ordinary workstation conditions, the experiment is conducted in the Computer Graphics Laboratory under normal lighting and working conditions with one exception. A dividing wall is placed behind the subject to eliminate the possibility of light reflecting of the monitor in the subject’s vision. Dim ambient illumination illuminated the display surface at about 20 lux, and there is a low level of background noise from other workers.

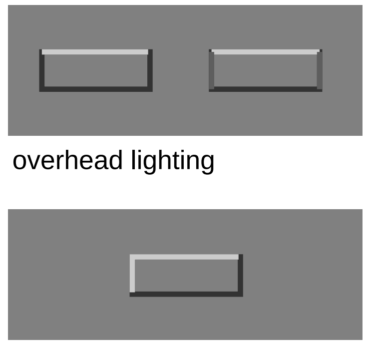
This research is intended to study the potential benefits of depth from shading cues for *practiced* users. It is assumed that with extended use of an interface using these cues, the users quickly become highly practiced, the state that this research addresses. Consequently, the subjects used are graduate students in the Computer Graphics Lab at the University of Waterloo. They are familiar with graphical interfaces, including several windowing systems. Thus, they are likely to exhibit suitable practiced behaviour.

4.2 Experiment 1: Effect of lighting direction

The basic component of a three-dimensionally shaded window is a flat rectangular shape. Existing window systems that employ shading have the directional light source in the upper left corner. This arrangement is reasonable from an aesthetic standpoint. All objects that are convex create the beveled edges by using a light coloured line on the top and left edges and a dark coloured line on the bottom and right edges. The flat facing plane is a medium colour. The light and dark areas are reversed for objects that are concave. This can be compared to either the unappealing three colour version of overhead lighting or the slightly more appealing four colour version (Figure 4-1). Given the aesthetic benefit of the upper left lighting, is there a functional benefit: does lighting from the upper left create three-dimensional representations that are perceived quickly, accurately, and effortlessly?

4.2.1 Subjects

The subjects were two Computer Graphics Lab graduate students (as described in the General Methodology Section). AP had no experience with visual search experimentation, whereas SL had a great deal of experience with visual search tasks, both as a subject and as an experimenter. Both subjects had a great deal of experience with graphical user interlaces, including window systems.



overhead lighting

lighting from the upper-left

Figure 4-1: Number of shades required for aesthetically pleasing shapes

Notice that for overhead lighting, the four shade button on the right looks more realistic than the three shade button on the left. Whereas for lighting from the upper-left the three shade button looks good.

4.2.2 Procedure

The stimuli are rectangular shapes resembling push buttons. These buttons are 25 pixels in height and 70 pixels in width. The target is a single concave rectangular shape (depressed button); the distractors are convex rectangular shapes (standard button which is not depressed). The grid location and amount of jiggle of the target is randomly chosen. Recall that the stimulus consists of either four or sixteen items. In addition to the basic visual search conditions (targets presence and number of display objects), this experiment manipulates the direction of the light source. The different light source directions used are overhead lighting (lighting from the top), lighting from the upper left, from the upper right, and from the extreme left (Figure 4-2). These directions are each presented in a separate experimental block of trials to allow subjects to be accustomed to the direction of lightsource without having to think about the direction of the lightsource for each trial; recall that each lightsource direction can appear to be from the opposite direction (with the apparent depth of the objects being reversed). This is consistent with the use of window systems; the apparent lightsource in a window system does not change over time. Each block consisted of 256 trials with practice trials preceding each block of trials.

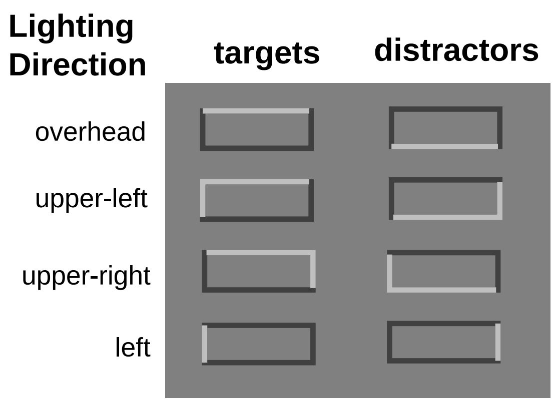


Figure 4-2: Targets and distractors for Experiment 1

Furthermore, for each trial, the button shapes in the grid were randomly jiggled in both the horizontal and vertical dimensions to prevent collinearity of the stimuli. For one subject (AP), the shapes were jiggled between -20 and +20 pixels. For another subject, the author (SL), the shapes were jiggled between -40 and +40 pixels.

4.2.3 Results

Each subject is analyzed separately. Overall accuracy for subject AP was 99.4%, which was quite constant across blocks. Overall accuracy for subject SL was 93.9%, with accuracy for the left lighting condition lower than for the other conditions, but not significantly. The slopes of the reaction times are analyzed with incorrect responses and outliers removed[1], where outliers are determined separately for each of the eight conditions (2 target presence/absence × 4 light source directions).

To determine if the slopes are significantly less than ten milliseconds, 95% confidence intervals were constructed around the slopes. A slope is significantly less than ten milliseconds if the upper bound of the confidence interval is less than ten milliseconds. For both subjects AP and SL, all slopes except for the target absent with lighting from the left are significantly less than ten milliseconds (Table 4-1). Furthermore, slopes are similar between the target-present and target-absent conditions for all lighting conditions except for left lighting, where target-present and target-absent slopes become very different.

Subject	Lighting	target present				target absent			
		inter-cept	b	s_b	95% CI *	inter-cept	b	s_b	95% CI *
AP	overhead	531	2.1	1.1	4.2	572	-0.2	1.3	1.9
	upper-left	526	1.9	1.3	2.3	539	-1.0	1.1	0.8
	upper-right	594	2.8	1.5	5.8	662	1.2	2.3	5.8
	left	613	3.4	1.5	6.4	814	22.6	5.0	32.6
SL	overhead	428	1.7	0.7	3.1	470	0.4	0.7	1.8
	upper-left	419	1.7	0.7	3.0	457	0.1	0.7	1.6
	upper-right	448	1.7	0.9	3.4	466	1.4	0.8	3.0
	left	592	1.0	1.9	4.8	598	7.1	1.5	10.0

Table 4-1: Effect of lighting direction results

As the upper bound of the 95% confidence interval around the slope (b) shows, all slopes were significantly less than ten milliseconds except for the left lighting condition with the target absent (indicated by both subjects). Upper bounds of the 95% confidence interval around the slopes greater than ten milliseconds are shaded.

*upper bound

1. Outliers are calculated in Data Desk (Velleman, 1989) as those scores greater than `high_hinge + 1.5 x (high_hinge - low_hinge)` and less than `low_hinge  - 1.5 x (high_hinge - low_hinge)`, where high_hinge is the median of the data from the median to the maximum and low_hinge is the median of the data from the minimum to the median.

These results indicate that lighting from the upper left works as well as lighting from overhead in terms of being a cue that can be perceived quickly, accurately, and effortlessly, yet lighting from the left is not as beneficial. These results are also consistent with the overhead lighting bias reported by other researchers: the depth from shading objects were not processed as quickly when lit from the side as when lit from above.

4.3 Experiment 2: Depth from shading cues added to windows

Having established that directional lighting from the upper left benefits the user for button objects, it can be investigated whether adding depth from shading cues to windows with upper left lighting also results in quick, accurate and effortless perception. This experiment primarily presented windows with different target cues, although the button object of Experiment 1 was added for comparison.

4.3.1 Subjects

The subjects were seven Computer Graphics Lab graduate students (as described in the General Methodology Section) and one professor. Subjects GV, IB, ED, FJ, and SM had no experience with visual search experimentation, subject JW had some experience, and SL and WC had a great deal of experience with visual search tasks, both as subjects and as experimenters. All subjects had a great deal of experience with graphical user interfaces, including window systems.

4.3.2 Procedure

This experiment consists of four different object types. For two of the window conditions, the distractor window objects are identical - only the appearance of the target varies. The targets use depth from shading cues; one has a concave titlebar area and the other has a concave input area. The third window target cue condition is the control condition. This condition uses the titlestring as the cue. The distractors present the word ‘requests’ in the titlebar and the target presented the word ‘selected’ in the titlebar. The button condition of Experiment 1 is also included. The target and distractor type for each condition is presented in Figure 4-3. Note that the window conditions are not completely analogous to the button condition wince the cue in the unselected window does not protrude up from the immediate background (the window frame) as does the button. However, this more accurately reflects the way that window systems are likely to be used. The stimuli windows are 80 pixels in height and 120 pixels in width. The concave titlebar cue is 14 pixels in height and 80 in width and the concave input area is 40 pixels in height and 101 in width. A two level text condition (text / no text) was also added to determine whether the addition of text to the windows affects the depth from shading cue perception; because windows are commonly used to display text, it is important to know that the depth from shading functional benefits are not masked when text is used in the windows.

As in Experiment 1, either four or sixteen elements were presented, on a 4×4 grid, randomly jiggled between -20 and +20 pixels in each of the horizontal and vertical dimensions. The object type conditions were presented in blocks with the text manipulation randomized within these blocks. Each block consisted of 512 trials with ten practice trials preceeding each block of trials. The block presentation order was counterbalanced across subjects. Also, the subjects have a short practice session before the start of the experiment consisting of 24 trials in each block.

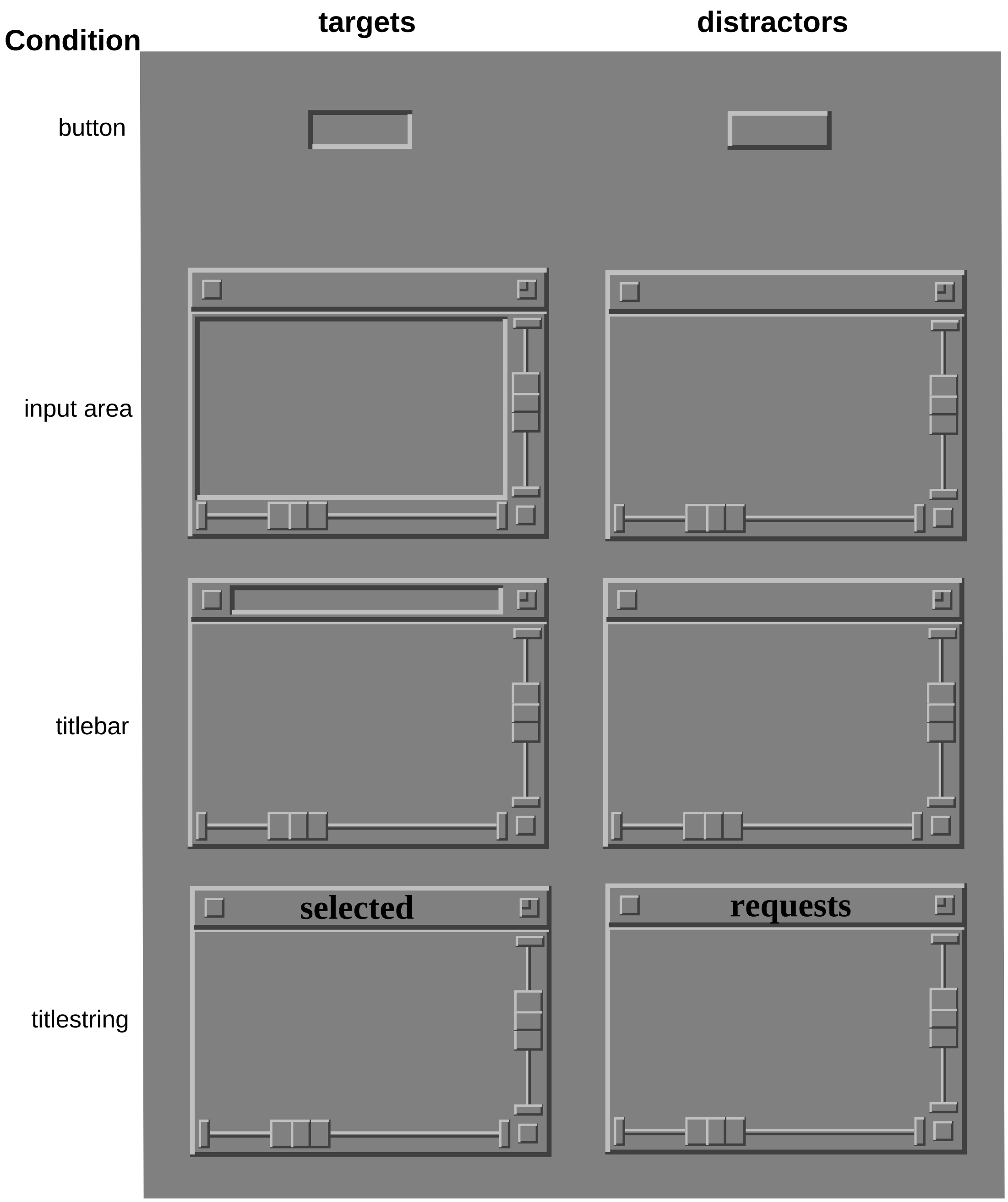


Figure 4-3: Targets and distractors for Experiment 2

	Subjects							
	JW	GV	IB	ED	FJ	WC	SM	SL
button	98.8	96.3	98.3	96.3	97.5	96.7	93.9	94.5
input area	98.4	95.7	96.1	95.3	98.2	94.3	91.6	91.8
titlebar	97.5	92.4	94.2	96.7	97.7	87.3	94.5	90.0
titlestring	96.5	94.5	86.6	92.5	91.2	89.1	89.5	91.4

Table 4-2: Accuracies for depth from shading cues added to windows
Five out of eight subjects have their lowest accuracies for the control condition (JW, IB, ED, FJ, SM). The remaining three subjects (GV, WC, SL) produced identical pattern, where the titlebar condition resulted in the lowest accuracies followed by the control condition.

4.3.3 Results

Results for each subject was analyzed separately. Overall accuracy for each of the subjects is presented in Table 4-2. Five out of the eight subjects (JW, IB, ED, FJ, SM) had the lowest accuracy for the control condition titlestring. This is as would be expected. A target text string that is similar in length and has similar types or proportions of characters (i.e., round characters such as ‘c’, ‘d’, and ‘q’ versus angular characters such as ‘l’ and ‘t’) to its distractors are very difficult to recognize. Thus, the control condition titlestring is likely to result in high reaction time slopes, low accuracy, or both. The remaining three subjects (GV, WC, SL) produced identical accuracy patterns; the button condition yielded the highest accuracies, the input area condition yielded the next highest accuracies, followed by the control condition, then the titlebar condition. These three subjects later proved to exhibit the most practiced behaviour. Practiced behaviour for effortless perception most likely uses the strategy that if the target is not identified immediately, then it is assumed to be absent. It is possible that these three subjects changed strategies between the depth from shading trials and the control (titlestring) trials, making a different trade-off between speed and accuracy. For the depth from shading conditions, processing could be done quickly and thus emphasis was made on quick processing. Whereas, for the control condition, processing could not be done quickly and thus emphasis was made on accurate responses. Although, it is difficult to verify from the data. The more effortless the perceptual processing, the more likely that both accuracy will be high and reaction time will be low. However, as the processing becomes slightly less effortless, either mistakes or reaction time will suffer or both. From the data, given that accuracy did not suffer, it is impossible to determine if reaction time increased more than it would have otherwise. Regardless, the accuracies were well within the range expected.

The slopes of the reactions times were analyzed with incorrect responses and outliers removed[2], where outliers were determined separately for each of the sixteen conditions (2 target presence/absence conditions × 2 text conditions × 4 objects types). Similar to Experiment 1, 95% confidence intervals were constructed around the slopes and a slope is significantly less than ten milliseconds if the upper bound of the confidence interval is less than ten milliseconds. The intercepts, slopes, standard error of the slopes, and upper bound of the interval for each of the conditions for each of the subjects is presented in Table 4-3.

Table 4-4 shows the general pattern. Consider the depth from shading experimental (button, input area, and titlebar) conditions. Not many subjects produced slopes greater than ten milliseconds for the target-present condition versus the target-absent condition. And this is true of both the text and no text conditions.Collapsing across the depth from shading and the text/no text conditions, eight subjects produced slopes greater than ten milliseconds for the target-present condition, whereas 23 subjects produced large slopes for the target-absent condition. Further, compare the depth from shading experimental conditions with the control (titlestring) condition. Not all subjects produced slopes greater than ten milliseconds in the depth from shading conditions (and in some cases there were few or none), whereas all subjects produced slopes greater than ten milliseconds in the control condition. Further, Table 4-3 indicates that in all cases the control condition produced slopes that were much greater than slopes

2. Outliers are defined the same as for Experiment 1 (Section 4.2.2)

Subject			target present				target absent			
			inter-cept	b	s_b	95% CI*	inter-cept	b	s_b	95% CI*
JW	text	button	550	6.7	2.1	10.8	737	18.7	2.8	24.2
		input area	553	4.1	2.0	8.1	880	19.0	3.8	26.4
		titlebar	571	8.2	2.6	13.4	836	28.6	3.5	35.7
		titlestring	750	110.8	12.0	134.5	567	237.1	8.7	254.4
	no text	button	522	3.2	1.5	6.2	655	14.2	2.7	19.7
		input area	557	3.3	1.9	6.4	788	27.3	4.0	35.1
		titlebar	559	5.2	2.2	9.6	752	28.8	3.2	35.2
		titlestring	703	112.4	12.0	136.3	507	236.3	8.3	252.8
GV	text	button	486	3.2	1.1	5.3	513	2.2	1.2	4.5
		input area	508	3.8	1.2	6.1	709	3.8	1.7	7.1
		titlebar	499	3.4	1.0	5.5	655	10.6	2.1	14.8
		titlestring	933	76.9	9.1	94.9	666	215.7	5.2	226.0
	no text	button	461	2.0	0.8	3.6	515	0.5	1.1	2.7
		input area	513	1.9	0.9	3.7	613	6.9	1.5	9.8
		titlebar	484	4.1	1.1	6.3	575	9.1	1.4	11.8
		titlestring	710	119.5	12.0	143.2	495	230.1	5.8	241.6

Table 4-3: Slopes for depth from shading cues added to windows

The upper bound of the 95% confidence interval around the slope, b, indicate which conditions for each subject were significantly less than ten milliseconds.Upper bounds of the 95% confidence interval around the slopes greater than ten milliseconds are shaded.

*upper bound

Subject			target present				target absent			
			inter-cept	b	s_b	95% CI*	inter-cept	b	s_b	95% CI*
IB	text	button	544	3.2	1.3	5.7	620	3.1	1.9	6.9
		input area	674	2.7	2.1	6.9	929	4.8	3.8	12.5
		titlebar	594	5.1	2.0	9.0	774	3.2	2.2	7.6
		titlestring	969	48.4	10.9	69.9	862	130.4	8.7	147.6
	no text	button	507	1.5	0.9	3.2	573	1.9	1.7	5.3
		input area	613	5.2	1.4	7.9	748	9.2	2.5	14.2
		titlebar	550	5.8	1.6	8.9	663	8.8	1.9	12.6
		titlestring	910	47.1	8.4	63.8	1011	109.4	9.2	127.5
ED	text	button	835	2.2	4.0	10.0	1022	13.8	5.6	25.0
		input area	861	5.8	4.2	14.1	1299	33.1	7.6	48.1
		titlebar	824	18.3	5.5	29.3	1162	59.7	7.9	75.3
		titlestring	1682	32.5	16.8	65.8	1325	242.0	17.9	277.8
	no text	button	754	1.8	3.4	8.5	879	5.6	4.0	13.6
		input area	806	11.4	4.0	19.4	1002	44.7	7.1	58.7
		titlebar	844	3.7	5.0	13.5	959	29.6	4.6	38.7
		titlestring	1313	96.0	17.1	129.8	1321	217.8	19.5	256.4

Table 4-3: Slopes for depth from shading cues added to windows (Continued)
The upper bound of the 95% confidence interval around the slope, b, indicate which conditions for each subject were significantly less than ten milliseconds.Upper bounds of the 95% confidence interval around the slopes greater than ten milliseconds are shaded.

*upper bound

Subject			target present				target absent			
			inter-cept	b	s_b	95% CI*	inter-cept	b	s_b	95% CI*
FJ	text	button	563	3.5	1.6	6.6	652	12.1	2.5	17.1
		input area	638	4.0	1.9	7.8	823	39.7	4.0	47.6
		titlebar	653	6.3	2.3	10.8	770	21.8	2.9	27.6
		titlestring	866	71.3	11.0	93.1	1053	195.1	8.7	212.3
	no text	button	540	0.9	1.3	3.5	641	1.9	1.6	5.1
		input area	639	3.9	1.9	7.7	811	25.9	3.6	33.1
		titlebar	599	1.5	1.4	4.3	692	14.3	2.4	19.2
		titlestring	939	101.8	15.2	131.9	888	194.3	10.0	214.2
WC	text	button	635	1.0	1.4	3.8	677	0.8	1.3	3.4
		input area	741	5.6	2.5	10.6	965	2.9	2.9	8.5
		titlebar	767	2.2	3.0	8.0	924	4.1	2.7	9.4
		titlestring	1123	57.8	10.3	78.2	702	222.6	8.3	239.0
	no text	button	580	1.9	0.9	3.7	651	-0.5	1.1	1.7
		input area	729	4.4	2.2	8.8	825	6.3	2.4	11.1
		titlebar	755	2.0	2.6	7.2	841	4.9	1.9	8.7
		titlestring	1086	71.0	11.7	94.2	772	209.4	9.3	227.8

Table 4-3: Slopes for depth from shading cues added to windows (Continued)

The upper bound of the 95% confidence interval around the slope, b, indicate which conditions for each subject were significantly less than ten milliseconds.Upper bounds of the 95% confidence interval around the slopes greater than ten milliseconds are shaded.

*upper bound

Subject			target present				target absent			
			inter-cept	b	s_b	95% CI*	inter-cept	b	s_b	95% CI*
SM	text	button	533	4.4	1.7	7.8	556	5.5	2.0	9.5
		input area	593	3.0	2.0	6.9	881	0.1	4.0	7.9
		titlebar	646	2.7	2.1	6.8	771	4.6	3.8	12.1
		titlestring	809	86.2	10.2	106.3	842	177.5	11.1	199.4
	no text	button	491	2.6	1.1	4.8	572	0.5	1.4	3.3
		input area	480	3.6	1.7	7.0	713	-1.0	2.1	3.3
		titlebar	576	3.1	1.4	5.9	703	2.3	2.6	7.3
		titlestring	867	94.0	11.5	116.6	687	187.5	8.0	203.2
SL	text	button	481	2.0	1.0	4.1	495	0.0	0.8	1.5
		input area	449	2.9	0.9	4.6	598	1.7	1.5	4.6
		titlebar	486	3.0	1.3	5.6	531	1.4	1.1	3.6
		titlestring	638	106.5	13.0	132.2	502	216.2	5.4	227.0
	no text	button	436	2.1	0.7	3.6	469	0.9	0.7	2.3
		input area	464	0.9	0.9	2.8	531	3.3	1.1	5.6
		titlebar	469	1.4	0.8	3.1	505	2.1	1.0	4.1
		titlestring	595	111.2	11.8	134.5	550	201.6	5.9	213.4

Table 4-3: Slopes for depth from shading cues added to windows (Continued)

The upper bound of the 95% confidence interval around the slope, b, indicate which conditions for each subject were significantly less than ten milliseconds.Upper bounds of the 95% confidence interval around the slopes greater than ten milliseconds are shaded.

*upper bound

of the depth from shading conditions.

A pattern also emerges with respect to the subjects. Six of the eight slopes greater than ten milliseconds for the target-present trials were produced by only two subjects. Further, almost two-thirds (17) of the twenty-four slopes greater than ten milliseconds for the target-absent trials were produced only three subjects, where two of these subjects were the same two that produced most of the large target-present slopes. Contrast this general performance trend with the performance for the control condition. All subjects produced slopes significantly greater than ten milliseconds for all conditions.

	text				no text			
	target present		target absent		target present		target absent	
button	J	1	J EF	3		0	J E	2
input area	E W	2	J IEF	4	E	1	J IEF	4
titlebar	J EF	3	JG EF S	5	E	1	JGIEF	5
titlestring	JGIEFWSS	8	JGIEFWSS	8	JGIEFWSS	8	JGIEFWSS	8

Table 4-4: Subjects producing slopes greater than ten milliseconds

First initial of each subject and the number of subjects who produced a slope not significantly less than ten milliseconds for each condition. Notice that the number of subjects is smaller for the experimental conditions than for the control condition (titlestring), in particular for the target present condition. Also notice that it is generally the same subjects who produce slopes greater than ten milliseconds for the experimental conditions.

From these results it seems that almost all can use the depth from shading cue to find the target: few subjects have slopes greater than ten milliseconds for the target present conditions of the depth from shading conditions button, input area, and titlebar. Note that the titlestring condition is the control condition. Some subjects (AP & SL for Experiment 1 (Table4-1); SL, WC, SM, GV, & IB for Experiment 2 (Table 4-3)) do not have slopes greater than ten milliseconds for either the target-present or the target-absent trials for all depth from shading conditions indicating practiced behaviour.

Practiced subjects use the strategy ‘if I don't see it right away, then it's not there’: target-present and target-absent slopes are similar; others (FJ & JW for Experiment 2 (Table 4-3)) are unpracticed and if they do not see it right away, they try to verify and look for it: slopes for target-absent trials are much greater than slopes for target-present trials. Given practice, the belief is that all subjects would adopt a non-verification strategy. Meanwhile, all subjects have difficulty finding the control target.

4.4 Experiment 3: Overlapping windows with depth from shading cues

Having established that adding depth from shading cues to windows with upper left lighting also results in quick, accurate and effortless perception, especially once the users become practiced, more realistic window systems can be studied. A more realistic window system allows windows to have a range of sizes, text may be in some windows and not others, and windows can be in any location, that is, the windows can overlap one another. This experiment is similar to the previous experiment except random variability was added to the number of windows displayed, whether text is present in the windows, and the window locations and sizes.

4.4.1 Subjects

To reiterate, it is intended that this research will study the potential benefits of depth from shading cues for practiced users. Therefore, two of the subjects that exhibited practiced behaviour were used in this third experiment. The subjects were Computer Graphics Lab members - one a graduate students and one professor. Both SL and WC had a great deal of experience with visual search tasks, both as subjects and as experimenters. And both subjects had a great deal of experience with graphical user interfaces, including window systems.

4.4.2 Procedure

The windows are displayed in a 4 × 3 grid-like arrangement to which the random variability was added. Each window had a 80 percent probability of being visible. Each window had an 80 percent probability of having text displayed in its content area. The size of each window was randomly chosen value between 60 and 400 for the width and 70 and 300 for the height. The location of each window was determined by adding a random jitter value between -70 and +70 each of the x and y values of the grid coordinate. Since this arrangement allows for overlap of windows, the order in which the windows were drawn were also randomized. The target window was randomly chosen from the visible windows. The target-absent trials were duplicates of the target-present trials with the notable exception of the absence of the target. The target window was identified by a concave titlebar region.

Similar to the previous experiments, the subject was to indicate quickly whether the target was present or absent by pressing the appropriate key while maintaining a high degree of accuracy. The experiment consisted of 400 trials with 16 practice trials preceding the experimental trials.

4.4.3 Results

For this experiment, only target presence was controlled; each display configuration was displayed twice during an experimental session - once for target present and once for target absent. Thus, target presence was analyzed for each subject with an ANOVA. For both subjects, accuracy for the target absent condition was perfect or near perfect with significantly lower accuracies for the target present condition ($F=39.94$, $p<0.001$, $df=199$ for SL; $F=21.33$, $p<0.001$, $df=199$ for WC). The accuracies for the target-absent and target-present conditions were 100% and 89% for SL and 99.5% and 82% for WC. This pattern, which is target-absent biased, is expected if a variation of the practiced strategy is used: the subject indicates that the target was not seen if a given amount of time has passed before the target is located, resulting in a speed-accuracy trade-off. By responding reasonably quickly with a target-absent bias, the accuracy for the target-present trials is reduced. When looking at the slopes of reaction time per element displayed, it appears that SL did produce a speed-accuracy trade-off, biased for speed. Both the target-present (slope=0.114) and the target-absent (slope=1.68) slopes were very small. However, this may not be the case for WC. Both target-present (slope=50.7) and target-absent (slope=121.1) slopes were very large *and* the target-absent slope was more than double the target-present slope.

Since all other variables had values that were randomized, ANOVAs are not appropriate (see Section 2.5). Thus, these variables were primarily analyzed using correlation analyses. Note that only target-present trials are analyzed since with the exception of the variable representing the number of windows, the variables to be analyzed represent some aspect of the cue: visible perimeter is defined as 2 × (height + width) of the cue in pixels; visible area is defined as (height × width) of the cue in pixels; distance from fixation is defined as the distance from the centre of the cue to the fixation point (centre of the screen) in pixels; text in target is a boolean variable representing whether text is present in the target window; number of corners visible counts the number of visible corners of the cue, where a visible corner is defined as two sides that meet where each side is greater than zero pixels in length; number of sides visible counts the number of visible sides of the cue, where a side is defined as visible if the whole side is completely visible from corner to corner; titlebar cue width represents the width of the titlebar cue in pixels. Thus, these variables are only meaningful for the target-present trials.

The correlation coefficients of the independent variables and correctness (and each other) are presented in Table 4-5 for SL and Table 4-6 for WC. The (darkly) shaded correlation coefficients in the first column are significant. Significant correlations with correctness are of primary interest, however, significant correlations between other variables help to explain the data. Thus, the (lightly) shaded correlation coefficients in the other columns are significant correlations between variables that are significantly correlated with correctness.

For subject SL, as illustrated in Table 4-5, visible area, number of corners visible, and titlebar cue width are positively correlated with correctness (*r*=0.295, *r*=0.352, and *r*=0.162, respectively, df=199): as the visible area increases, accuracy increases; as the number of visible corners increases, accuracy increases; as the titlebar cue width increases, accuracy increases. Note that visible area and number of visible corners, as well as visible area and titlebar cue width are significantly positively correlated

	correctness	number of windows	visible perimeter	visible area	distance from fixation	text in target	number of corners visible	number of sides visible
number of windows	0.004							
visible perimeter	0.052	-0.120						
visible area	0.296	-0.079	-0.084					
distance from fixation	0.028	-0.011	0.169	-0.106				
text in target	0.042	0.144	-0.023	-0.088	-0.061			
number of corners visible	0.352	-0.067	0.340	0.170	0.065	-0.029		
number of sides visible	0.080	-0.095	0.470	0.100	0.059	-0.092	0.801	
titlebar cue width	0.162	-0.019	-0.338	0.732	-0.093	-0.010	-0.298	-0.391

Table 4-5: Correlations with correctness for subject SL

Darkly shaded cells (in the first column) indicate a significant correlation with correctness (significant variables); lightly shaded cells (in the other columns) indicate a significant correlation between significant variables. All these correlational analyses have 199 degrees of freedom. Note that the smallest significant correlation is 0.138.

(*r*=0.170 and *r*=0.732, respectively df=199). This means that it is possible that visible area is truly correlated with correctness and it is the correlation of the other variables with visible area that causes their correlations with correctness. Or, conversely, one or both of number of visible corners and titlebar cue width are truly correlated with correctness and it is the correlation of visible area with these variables which causes the correlation of visible area with correctness.

Also note that number of visible corners is negatively correlated with titlebar cue width (*r*=-0.298, df=199). Given this negative correlation, in conjunction with the positive correlations of both number of visible corners and titlebar cue width with correctness, then it is likely that one of these relations is spurious (see Section 2.5).

For subject WC, as illustrated in Table 4-6, visible perimeter, visible area, text in

	correctness	number of windows	visible perimeter	visible area	distance from fixation	text in target	number of corners visible	number of sides visible
number of windows	-0.128							
visible perimeter	0.311	-0.098						
visible area	0.230	-0.057	0.170					
distance from fixation	-0.104	-0.091	0.094	-0.055				
text in target	0.194	-0.097	-0.042	0.128	-0.028			
number of corners visible	0.521	-0.125	0.478	0.349	-0.102	0.137		
number of sides visible	0.432	-0.118	0.457	0.337	-0.109	0.129	0.921	
titlebar cue width	0.001	-0.011	-0.083	0.757	0.003	0.086	-0.179	-0.206

Table 4-6: Correlations with correctness for subject WC

Darkly shaded cells (in the first column) indicate a significant correlation with correctness (significant variables); lightly shaded cells (in the other columns) indicate a significant correlation between significant variables. All these correlational analyses have 199 degrees of freedom. Note that the smallest significant correlation is 0.138.

target, number of visible corners, and number of sides visible are each positively correlated with correctness (*r*=0.311, *r*=0.230, *r*=0.194, *r*=0.521, and *r*=0.432, respectively, df=199).

Notice that number of visible corners and number of sides visible is *strongly* correlated (*r*=0.921, df=199). Furthermore, notice that the correlations between visible area and each of number of visible corners and number of sides visible are similar (*r*=0.349 and *r*=0.337, df=199). And the correlations between visible perimeter and each of number of visible corners and number of sides visible are similar (*r*=0.478 and *r*=0.457, df=199). Therefore, it is likely that number of visible corners and number of sides visible is measuring the same relationship.

Also notice that visible perimeter and visible area are positively correlated (*r*=0.170, df=199). Thus, it is possible that not all of the variables are truly correlated with correctness.

When taking results from both SL and WC into account, they have in common positive correlations of visible area with correctness and number of visible corners with correctness. From these two correlations, each of the other correlations can be explained with the exception of text in target for WC. Furthermore, for both subjects, visible area and number of visible corners are highly correlated. It is important to note that the number of windows displayed was not correlated with accuracy for either subject, which agrees with the results of Experiment 2. Furthermore, it is important to keep in mind that the two subjects did not see the same experimental trials because this experiment was fully randomized except for the fixed variable of target presence.

The correlation coefficients of the variables and reaction time (and each other) are presented in Table 4-7 for SL and Table 4-8 for WC. Only data for correct responses for the target present trials are analyzed. Also, outliers[3] were removed from this data. Again, the shaded correlation coefficients are significant. Note that the signs of the correlations with percentage correct (Tables 4-5 and 4-6) are opposite that of the correlations with reaction time (Tables 4-7 and 4-8). This is because accuracy and reaction

3. Outliers are defined the same as for Experiment 1 (Section 4.2.2)

time are opposing dependent variables: as accuracy increases, reaction time decreases, and vice versa. The percentage correct matrices also differ from the reaction time matrices because the correct matrices include all the data, whereas the reaction time matrices have the incorrect responses and the outliers removed.

For subject SL, as illustrated in Table 4-7, visible area, number of visible corners, and number of sides visible are each significantly negatively correlated with reaction time (*r*=-0.220, *r*=-0.276, and *r*=-0.205, respectively, df=171). Also notice that number of visible corners and number of sides visible are *strongly* positively correlated (*r*=0.910, df=171). Thus, it is quite likely that these variables are measuring the same relationship.

	reaction time	number of windows	visible perimeter	visible area	distance from fixation	text in target	number of corners visible	number of sides visible
number of windows	0.001							
visible perimeter	0.066	-0.139						
visible area	-0.220	-0.110	-0.122					
distance from fixation	-0.094	0.001	0.132	-0.129				
text in target	0.057	0.142	-0.074	-0.121	-0.089			
number of corners visible	-0.276	-0.092	0.423	0.087	0.036	-0.041		
number of sides visible	-0.205	-0.099	0.424	0.111	0.047	-0.061	0.910	
titlebar cue width	-0.033	-0.038	-0.326	0.754	-0.137	-0.046	-0.416	-0.356

Table 4-7: Correlations with reaction time for subject SL

Darkly shaded cells (in the first column) indicate a significant correlation with reaction time (significant variables); lightly shaded cells (in the other columns) indicate a significant correlation between significant variables All these correlational analyses have 171 degrees of freedom. Note that the smallest significant correlation is 0.15.

As illustrated in Table 4-8, WC also produced similar results. Visible area, number of visible corners, and number of sides visible are each significantly negatively correlated with reaction time (r=-0.289, r=-0.362, and r=-0.324, respectively, df=171). Also, the number of visible corners is *strongly* positively correlated with number of sides visible (r=0.931, df=171) again, suggesting that it is quite likely that these variables are measuring the same relationship. Further, visible area is positively correlated with number of visible corners indicating the possibility that not both of these variables are truly correlated with reaction time (r=0.262, df=171). An important difference and key correlation is the correlation between the number of windows displayed

	reaction time	number of windows	visible perimeter	visible area	distance from fixation	text in target	number of corners visible	number of sides visible
number of windows	0.203							
visible perimeter	-0.098	-0.036						
visible area	-0.289	-0.036	0.092					
distance from fixation	0.050	-0.131	0.121	0.020				
text in target	-0.055	-0.087	-0.055	0.081	0.071			
number of corners visible	-0.362	-0.089	0.403	0.262	-0.031	0.081		
number of sides visible	-0.324	-0.050	0.387	0.249	-0.062	0.035	0.931	
titlebar cue width	-0.054	-0.029	-0.098	0.796	0.072	0.053	-0.242	-0.255

Table 4-8: Correlations with reaction time for subject WC

Shaded cells in the first column indicate a significant correlation with reaction time (significant variables); shaded cells in the other columns indicate a significant correlation between significant variables. All these correlational analyses have 171 degrees of freedom. Note that the smallest significant correlation is 0.15.

and reaction time (*r*=0.203, df=171). Furthermore, number of windows is not correlated with any other variable. This means that it is likely that as the number of windows increased, the reaction time also increased.

When taking results from both SL and WC into account, they have in common negative correlations of visible area, number of visible corners, and number of sides visible with reaction time. And both show strong correlations between number of visible corners and number of sides visible, indicating that they are likely measuring the same relationship. The subjects differ in the crucial correlation of the number of windows and reaction time.

The absence of a correlation between the number of windows and reaction time for subject SL is in keeping with the results from the previous experiment (with non-overlapping windows). That is to say, reaction time does not increase with an increasing number of windows displayed. A linear regression analysis of reaction time regressed on number of windows results in a slope of 0.114 milliseconds. Although this is virtually a zero milliseconds slope, the upper bound of the 95% confidence interval is 13.84 milliseconds. The fact that the slope is greater than ten milliseconds is due to the large standard error of the slope, which is the result of the variability introduced by the other variables (such as, visible area and number of visible corners). Subject WC, on the other hand, showed a positive correlation between the number of windows and reaction time. A linear regression analysis of reaction time regressed on number of windows results in a large slope of 50.678 milliseconds for the target-present condition and also a large slope of 121.1 milliseconds for the target-absent condition. This would indicate that for this task WC exhibited behaviour indicative of unpracticed subjects. This regression in behaviour from Experiment 2 to Experiment 3 is common when the experimental design is changed from a blocked design to a random design since it is easier to become practiced when similar stimuli are presented than when varying stimuli are presented.

The results of subject SL indicates that it is possible to process overlapping windows with depth cues effortlessly. Since subject WC produced slopes less than ten milliseconds for the previous experiment (with non-overlapping windows), but not for this (overlapping windows) experiment, it is possible that more practice is necessary to acquire effortless perception of depth cues in overlapping windows. Over the course of the development of the series of experiments SL had more practice than any other subject.

Chapter 5

Discussion and Future Work

The series of visual search experiments was conducted to explore whether adding shape from shading depth cues to window interfaces would make the display more understandable. Most window interfaces display output in sets of rectangular regions, windows, comprised of the canvas or input area, titlebar region, and various buttons and scroll bars. Previous research, reported in the psychology literature, indicated that the perception of depth conveyed by shading is processed preattentively, and in parallel. That is to say, concavity and convexity is perceived quickly, accurately, and effortlessly. However, most of the shapes used in psychological studies are smoothly varying gradients such as egg and egg-crate shapes. Window systems, on the other hand, use rectangular regions with beveled edges, which lack smoothly varying gradients, using discrete planes of different shade values to convey bevelled edges of the windows and its decorations.

The experiments presented in this thesis were designed to test the hypothesis that windows composed of rectangular bevelled shapes can be perceived quickly, accurately, and effortlessly. This hypothesis was tested by three supporting experiments.

5.1 Discussion of Experimental Results

The first tested the direction of the light source. Existing window systems that appear three-dimensional are generally presented with a directional light source from the upper-left corner. Previous research reported that in terms of perceiving depth, there is a bias toward overhead lighting (Brewster, 1826; Ramachandran, 1988a, 1988b) - objects are seen with greater depth if lit from above as opposed to being lit from the side - therefore, the parallel perception of depth disappears if the apparent light source is from the left or right. The results of this experiment supported the hypothesis that shaded representations of rectangular bevelled objects lit from the upper-left provide effortless perception of depth.

The second experiment tested is such rectangular bevelled depth cues can provide quick and effortless determination of the active window and if the presence of text in the window affects this perception. The presence of text in the windows only slightly decreased the instances of effortless perceptions. The effective use of depth cues was supported by almost all subjects. These subjects seemed to fall into one of two categories: practiced and unpracticed. Two subjects exhibited unpracticed behaviour: if the target was not seen ‘instantly’, the subjects tried to verify its absence by continuing to search for it. The remaining five subjects exhibited practiced behaviour: if the target was not seen ‘instantly’, then the subjects reported its absence without unnecessary verification. Presumably, with enough practice, any subjects will non-verification strategy. Thus, depth conveyed by shading is a good attentional cue in displays like window systems. Because depth is a two state cue it works for identifying the odd element in a well demarcated set: the set radio button or the active window, for example. Furthermore, the presence of text in the windows only slightly decreases the instances of effortless perception, likely because text and shape from shading have different characteristics.

The third experiment was designed to evaluate the effectiveness of the depth cues in a stimulus closer to a real window system, with windows varying in size, location, and overlap. The resulting depth cues can be partially occluded. Since we are primarily interested in the potential benefits to practiced users, two of the practiced subjects from the preceding experiment were used in this experiment. The results from one subject indicated that accuracy and reaction time were unaffected by the number of windows displayed. The fact that the number of windows was not significantly correlated with either correctness or reaction time is interpreted in support of the hypothesis that windows with depth cues can be perceived effortlessly, given the results from the second experiment. However, both accuracy and reaction time varied with other factors, such as visible area of cue and number of corners visible.

The other subject also produced results indicating that both accuracy and reaction time varied with other factors, such as visible area of target cue, number of corners visible, and number of sides visible, although in this case, number of corners visible and sides visible are likely to be measuring the same relationship. This subject, however, showed a relationship between the number of windows and reaction time. It is likely that more practice is required to achieve the performance of the first subject. Even though both subjects were chosen for their practiced behaviour, the first subject had significantly more practice than the other subject.

5.2 Generalizing to Window Systems

These results are encouraging for indicating the active window in a windowing system. The experiments were conducted with the assumption that the users of such a system will be practiced in the perception of the target cue. Even with the introduction of such an interface, where users have never used a interface in this manner, the results of these experiments indicate that most users will be able to perceive the target cue immediately if the cue is wholly visible. Thus, depth conveyed through shape from shading as an attentional cue can definitely be used by a system, such as Macintosh, that makes the active window also the top window. Users may not necessarily assume immediately that the active window cue is not visible if it is not seen instantly. Although the ability to recognize that the active window cue is not present if it is not perceived immediately will come with use of the system (use of the system becomes perception practice), the absence of this practiced behaviour is not entirely deficient. In a window system where the attentional cue is never occluded, if the active window is not perceived or located instantly, then most likely there is no active window. Thus, the user will have to make another window active. Furthermore, most have one window active at all times. Thus, target-present trials are most relevant for ‘real’ window systems.[1]

The results of the last experiment are also encouraging. The results indicate that it is possible to use depth conveyed through shape from shading as an attentional cue in a windowing system that allows overlapping windows. However, it is likely that more practice will be needed. But, of course, this practice can be acquired with continued use of the system. Graphical interfaces should and commonly do incorporate redundancy into the interface to ensure proper perception. Redundancy would particularly benefit users who have not yet reached the expert or practiced state.

Moreover, the depth cue in the titlebar was used for the third experiment, yet in the second experiment, the depth cue in the canvas input area produced better results. Per-

1. It is possible to have no window active in a Macintosh-style system. If the current application does not have a window, then all windows (belonging to inactive applications) will be inactive.

haps the depth cue in the input area would fair better in an overlapping window system - or better yet, both depth cues in the titlebar area and in the input area together.

5.3 Demonstration Application

A demonstration application was created to illustrate how a window system could look and function using shape from shading as an active window cue. Windows are created by selecting New Window from the menu. These windows can be modified in usual ways: clicking on the close box closes the window; clicking and dragging the scale box scales the window; clicking on the zoom box toggles the window size between the original standard window size and the user defined size, defined by the operation on the scale box; clicking on a window makes it active and brings it to the top of the window stack. In this demonstration window system, the active window has both a concave titlebar area and a concave canvas input area. If the demonstration application is made inactive (that is, another application, such as the Finder, is make active), then the active window does not need to be selected by a mouse click to be made active, merely moving the mouse over a window will cause it to become active. However, it will not be moved to the top of the window stack unless it is selected by a mouse click. This interface is similar to any X-Window windowing system, such as Motif, for example. Window activation through an idle mouse task illustrates how this type of active window cue could be used for a window system that allows this type of interaction, such as Motif.

Consider the following common situation in window system use. The user is primarily working (say, editing her thesis) in one window, which is fairly large. She needs to look at information in another window while performing the text editing task (see Figure 5-1). If the window system only allows the top window to be active, similar to

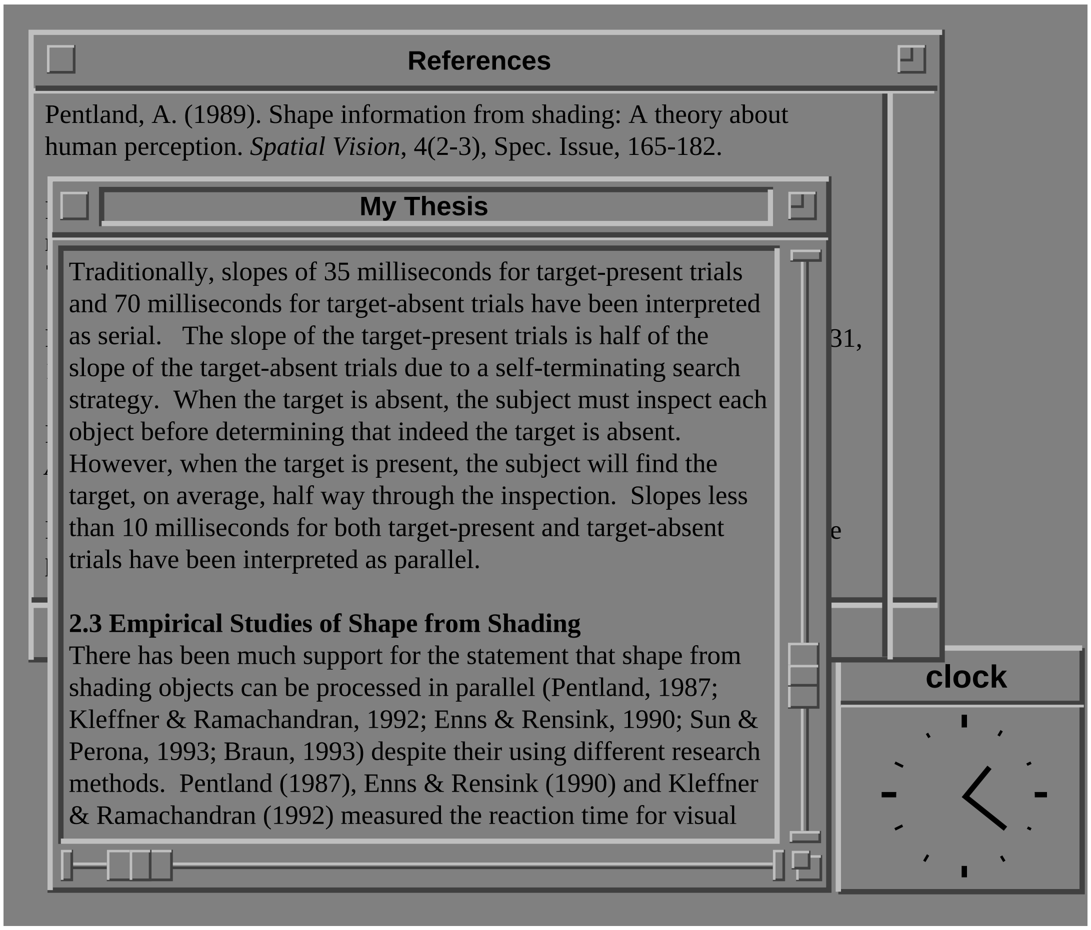


Figure 5-1: Top window is active

If the window system only allows the top window to be active, then difficult situations can arise. For instance, when the user wishes to see another window which is obscured by a large top window, but does not want to change the active focus, then the user will have to resize the top window.

the Macintosh windowing system, for example, the thesis window will obscure the other (references) window. Thus, she will have to resize the window that contains her thesis in order to see the References window. However, if the window system does not insist that the active window is the top window, then the configuration in Figure 5-2 is

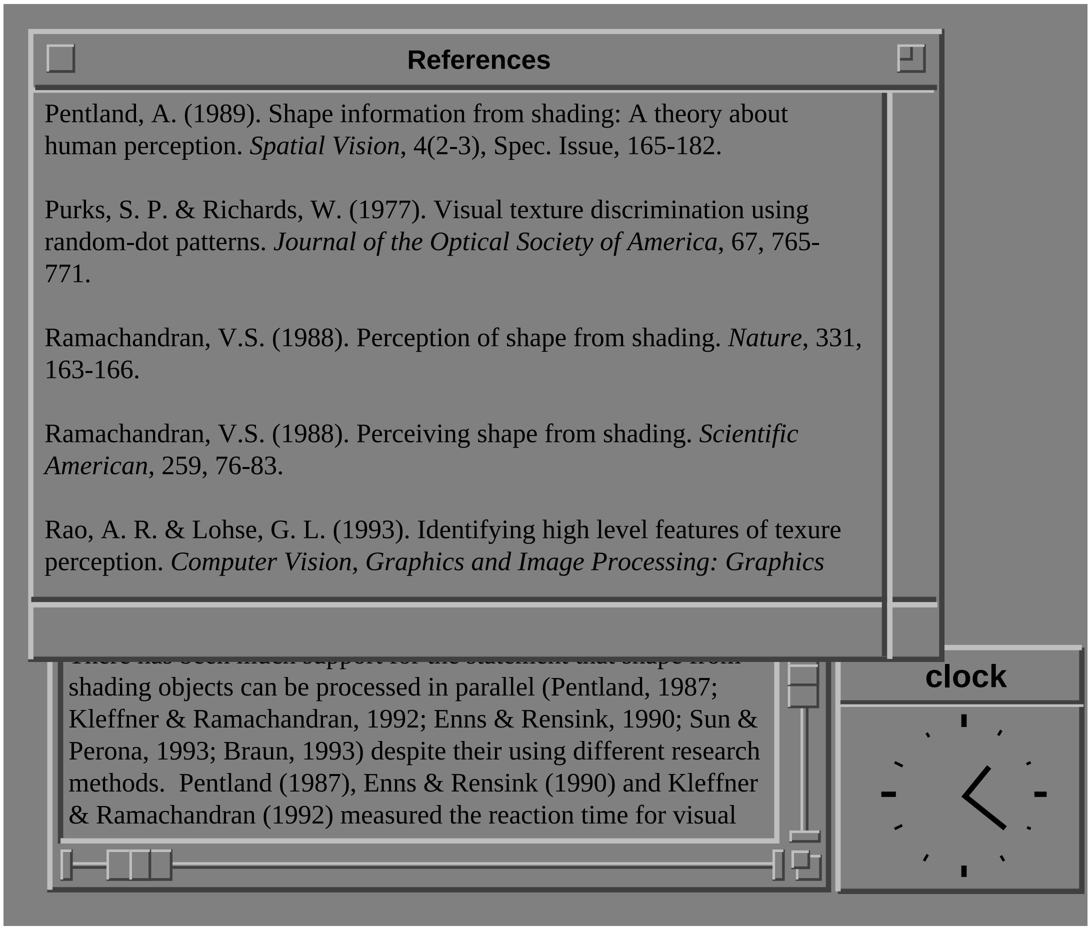


Figure 5-2: Active window is not on top

Even though the active window is not on top and the titlebar is obscured, the active window cues are still visible.

possible and no resizing is necessary. Using the concave canvas input area cue the user can perceive which window is active even though the whole window nor the titlebar is visible.

5.4 Generalizing to other User Interfaces

The results of this experiment in conjunction with results from other studies (Kleffner & Ramachandran, 1992; Braun, 1993) indicate that depth from shading is an effective cue in suer interfaces for drawing attention to a particular location in the interface. The experiments presented here explicitly tested and supported the hypothesis that depth from shading can be used effectively as a cue to indicate the active window in a windowing system. Also, implicit in the button condition of Experiment 2, the hypothesis that depth from shading can be an effective cue in form-based interfaces to indicate the active field was tested and supported. A form-based interface must indicate to the user which field is accepting input from the keyboard. So by using depth from shading cues, the currently active field in the form appears concave while all others appear convex (or even flat). The button condition of Experiment 2 presented similar features: the target or active field was concave and all other distractors or inactive fields were convex. Moreover, text in the filed (or button) objects did not affect the performance of the subjects.

The experiments presented here used rectangular cube (or button) objects or compositions of these objects, in the case of window objects. The effortless perception of these shapes indicate that the depth from shading cue can be used in any interface that requires these cube objects. Also, given the results of previous studies - effortless perception of smooth gradient (egg and egg-crate) objects (Kleffner & Ramachandran, 1992; Braun, 1993) - depth from shading cues can be used effectively in any interface that requires these types of shapes, such as radio buttons, for example. The purpose of a radio button cluster is to select or indicate selection of one item from a set of items. Thus, in the depth from shading cue paradigm, the selected radio button would be concave and all others convex.

Since depth from shading provides two states (concave and convex) which is perceived quickly, accurately, and effortlessly, the depth cue can be used to differentiate one (or a few) elements in an interface from the rest.

5.5 Future Work

From this research, it is apparent that there may be a few consistent factors that correlate with accuracy and reaction time for perception of depth cues indicating the active window in an overlapping window system: visible area of the cue, number of visible corners of the cue, and number of sides visible of the cue. Thus, it seems reasonable that designers of windowing systems could incorporate what is known about these factors into the interface.

Schlueter (1990) created a window system called Jostling Windows, where misperceptions could be avoided by dynamically jostling the windows around. Schlueter designed Jostling Windows to disambiguate arrangements such as alignment of window borders, apparent spanning of objects across two windows, and apparent spanning of text across two windows when their baselines became aligned. This type of jostling windows system could be extended to ensure effortless and accurate perceptions of the active window depth from shading cue. The window system could jostle the windows if at least two corners of the depth from shading cue were not visible or if a certain minimum number of pixels of the depth from shading cue were not visible.

Another interesting question concerns the interaction of attentional cues. If depth from shading is used to indicate the active window and if colour is used to group and differentiate windows based on task, how does one attentional cue affect the perception of the other attentional cue? It is assumed that since both of these attentional cues are voluntary then the user can choose which cue to attend to and thus there will be no interaction. Yet it is important to study empirically.

Furthermore, this same question of interaction of attentional cues can be made of visual texture and depth from shading cues. In this case, it becomes an even more interesting question. Since visual texture and depth from shading are both portrayed using shading techniques, it is possible that some interaction may result. In fact, one could claim that depth from shading is a special case of visual texture, where they differ in repetition and granularity. Shape for depth from shading is singular and usually large compared to visual texture which is a repeated pattern of smaller elements. However, a depth from shading element could be repeated ina pattern to form a visual texture.

References

Amadasun, M. & King, R. (1989). Textural features corresponding to textural properties. *IEEE Transactions on Systems, Man, and Cybernetics*, 19, 1264-1274.

Beck, J. (1966a). Perception grouping produced by changes in orientation and shape. *Science*, 154, 538-540.

Beck, J. (1966b). Effect of orientation and of shape similarity on perceptual grouping. *Perceptual Psychophysics*, 1, 300-302.

Beck, J. (1967). Perceptual grouping produced by like figures. *Perceptual Psychophysics*, 2, 491-495.

Beck, J., Sutter, A., & Ivry, R. (1987) Spatial frequency channels and perceptual grouping in texture segregation. *Computer Vision, Graphics and Image Processing*, 37, 299-325.

Berbaum, K., Bever, T. & Chung, C. S. (1983). Light source in the perception of object shape. *Perception*, 12, 411-416.

Braun, J. (1993). Shape from shading is independent of visual attention and may be a ‘texton’. *Spatial Vision*, 7(4), 275-3.

Brewster, D. (1926). On the optical illusion of the conversion of cameos into intaglios and of intaglios into cameos, with an account of other analogous phenomena. *Edinburgh Journal of Science*, 4.

Brewster, D. (1932). *Letters on Natural Magic*. London: John Murray.

Brewster, D. (1847) On the conversion of relief by inverted vision. *Edinburgh Philosophical Transactions*, 15, 657.

Broadbent, D.E. (1958). *Perception and communication*. London: Pergamon.

Carter, R. C. (1982) Visual search with color. *Journal of Experimental Psychology: Human Perception and Performance*, 8(10), 127-136.

de la Croix & Tansey (1986) *Gardner’s Art Through the Ages* (8th edition). Orlando, Florida: Harcourt Brace Jovanovich Publishers.

Duncan, J. (1989) Boundary conditions on parallel processing in human vision. *Perception*, 18(4), 457-469.

Enns, J.T. & Rensink, R.A. (1990). Influence of scene-based properties on visual search, *Science*, 247, 721-723.

Fechner, G. T. (1860). Elemente der Psychophysik. (Adler, H. E.: trans.), *Elements of Psychophysics*. Holt, Rinehart & Winston.

Foley, J. D., van Dam, A., Feiner, S. K., & Hughes, J. F. (1990). *Computer Graphics*

Principles and Practice. Chapter 21: Animation, 1064-1065. Addison-Wesley Publishing Company.

Georgeson, M. A. (1979). Random-dot stereograms of real objects: Observations on stereo faces and moulds. *Perception*, 8, 585-588.

Green, B. F. & Anderson, L. K. (1956) Color coding in a visula search task. *Journal of Experimental Psychology*, 1, 19-24.

Gregory, R. L. & Gombrich, E. H. (1973). *Illusion in Nature and Art*. London: Gerald Duckworth & Company, Ltd.

Harvey, L. 0. & Gervais, M. J. (1978). Visual texture perception and Fourier analysis. *Perceptual Psychophysics*, 24, 534-542.

Harvey, L. 0. & Gervais, M. J. (1981). Internal representation of visual texture as the basis for the judgement of similarity. *Journal of Experimental Psychology: Human Perception and Performance*, 7, 741-753.

Howard, I. P. (1983). Occluding edges in apparent reversal of convexity and concavity. *Perception*, 12, 85-86.

Howard, I. P., Bergstrom, S. S. & Ohmi, M. (1990). Shape from shading in different frames of reference. *Perception*, 19, 523-530.

Howell, D. C. (1992). *Statistical Methods for Psychology* (3rd Ed.) Belmont:Wadsworth Publishing Company.

Julesz, B. (1981). Textons, the elements of texture perception and their interactions.

Nature, 290(12), 91-97.

Julesz, B. (1986). Texton gradients: The theory revisited. *Biological Cybernetics*, 54, 245-251.

Kahneman, D. (1973). *Attention and Effort*. New Jersey: Prentice-Hall.

Kleffner, D.A. & Ramachandran, V.S. (1992). On the perception of shape from shading. *Perception & Psychophysics*, 52(1), 18-36.

Koenderink, J. J. (1984). What does the occluding contour tell us about solid shape? *Perception*, 13, 321-330.

Koenderink, J. J. & van Doorn, A. J. (1982). The shape of smooth objects and the way contours end. *Perception*, 11,129-132.

Levine, M. W. & Shefner, J. M. (1991). *Fundamentals of Sensation & Perception* (2nd edition). Pacific Grove: Brooks/Cole Publishing Company.

Lin, Louis. (1993). Using visual texture in window systems. Master's thesis, University of Waterloo. Available as Research Report CS-93-58 from the Department of Computer Science.

MacIntyre, B. (1991) A constraint-based approach to dynamic colour management for windowing interfaces. Master's thesis, University of Waterloo. Available as Research Report CS-91-55 from the Department of Computer Science.

Nagy, A. l. & Sanchez, R. R. (1992) Chromaticity and luminance as coding dimensions in visual search. *Human Factors*, 34(5), 601-614.

Nagy, A. l. & Sanchez, R. R. (1990) Critical color differences determined with a visual search task. *Journal of the Optical Society of America A*, 7(7), 1209-1217.

Neisser, U. (1967). *Cognitive Psychology.* Appleton-Century-Crofts, New York.

Oppel, J. J. (1856). Uber ein Anaglyptoskip. *Annalen der Physik und Chemie*, 175, 466-469.

Osborne, Jamie & Thomas, Deanna. (1993) Working in the third dimension, *develop: The Apple Technical Journal*. 15, 103-114.

Pentland, A. (1982). Finding the illuminant direction.*Journal of the Optical Society of America*, 72, 448-455.

Pentland, A. (1987). Toward an Ideal 3-D CAD System. *Image Understanding and the Man-Machine Interface,* SPIE 758, 165-171.

Puerta, A. M., Guttman, E. F. & O'Day, T. (1989). Shadows as a binocular depth cue. *Investigative Ophthalmology and Visual Science*, 32 (ARVO Abstract), 252.

Purks, S. P. & Richards, W. (1977). Visual texture discrimination using random-dot patterns. *Journal of the Optical Society of America*, 67, 765-771.

Ramachandran, V.S. (1988a). Perception of shape from shading. *Nature*, 331, 163-166.

Ramachandran, V.S. (1988b). Perceiving shape from shading. *Scientific American,* 259, 76-83.

Rao, A. R. & Lohse, G. L. (1993). Identifying high level features of texture perception.

Computer Vision, Graphics and Image Processing: Graphics Models and Image Processing, 55(3), 218-233.

Reingold, E. M. & Hansen, W. J. (1945). *Data Structures in Pascal*. Boston: Little, Brown Company.

Rittenhouse, D. (1786). Explanation of an optical deception. *Transactions of the American Philosophical Society*, 2, 37-42.

Schlueter, K. (1990). Perceptual Synchronization in Window Systems. Master's thesis, University of Waterloo. Available as Research Report CS-90-36 from the Department of Computer Science.

Schneider, W. & Shiffrin, R.M. (1977). Controlled and automatic human information processing: I. Detection, search and attention. *Psychological Review*, 84, 1-66.

Schroder, H. (1858). Uber eine optische inversion bei betrachtung verkehrter, durch optische vorrichtung entworfener physischer bilder. *Annalen der Physik und Chemie*, 181, 298-311.

Scott, D. (1991). *Human-Computer Interaction: A cognitive ergonomics approach*. (Chapter 3: Introduction to Cognitive Ergonomics, 55-73). Ellis Horwood.

Smallman, H. S. & Boynton, R. M. (1990) Segregation of basic colors in an information display. *Journal of the Optical Society of America A*, 7(10), 1985-1994.

Smith, S. L. (1962) Color coding and visual search. *Journal of Experimental Psychology*, 64(5), 434-440.

Sun, J.Y. & Perona, P. (1993). Preattentive perception of elementary three-dimensional shapes. *Investigative Ophthalmology & Visual Science*, Suppl, 34.

Tamura, H., Mori, S., & Yamawaki, T. (1978). Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man, and Cybernetics*, 8, 460-473.

Thayer, G. H. (1909). *Concealing Coloration in the Animal Kingdom*. New York: MacMillan.

Todd, J. T. & Mingolla, E. (1983). Perception of surface curvature and direction of illumination from patterns of shading. *Journal of Experimental Psychology: Human Perception and Performance*, 9(4), 583-595.

Triesman, A. (1985). Preattentive processing in vision. *Computer Vision, Graphics, and Image Processing*, 34, 156-177.

Treisman, A. (1986). Features and objects in visual processing. *Scientific American*, 246, 106-115.

Townsend, J. T. (1990). Serial vs. Parallel processing: Sometimes they look like Tweedledum and Tweedledee but they can (and should) be distinguished. *Psychological Science*, 1(1), 46-54.

Velleman, P. F. (1989). *Data Desk Handbook*. Illinois: Odesta Corporation, Appendix 8A.

Ware, C. & Knight, W. (1992). Orderable dimensions of visual texture for data display: Orientation, size and contrast. In *CHI'92 proceedings*, 203-209.

Wenderoth, P. & Hickey, N. (1993). Object and head orientation effects on symmetry perception defined by shape from shading. *Perception*, 22, 1121-1130.

Yonas, A., Kuskowski, M. & Sternfels, S. (1979). The rold of frames of reference in the development of responsiveness to shading information. *Child Development*, 50, 495-500.

Appendix

The data results files of the experiments in this thesis can be found in the Computer Science Department's anonymous ftp site. The directions for accessing these files are as follows.

ftp cs-archive.uwaterloo.ca
name: anonymous
password: <your userid>
cd cs-archive
get README

The README file provides instructions for accessing all online technical reports as well as these data files.