Cognitive Engineering Laboratory

The Design and Evaluation of Transparent User Interfaces: From Theory to Practice

Beverly L. Harrison

CEL 96-10

Cognitive Engineering Laboratory Department of Mechanical & Industrial Engineering University of Toronto
5 King's College Rd. Toronto, Ontario, Canada M5S 3G8
Phone: (416) 978-7399 Fax: (416) 978-3453
Email: benfica@mie.utoronto.ca URL: www.ie.utoronto.ca/IE/HF/CEL/homepage.html

Cognitive Engineering Laboratory

Director: Kim J. Vicente, B.A.Sc., M.S., Ph.D.

The Cognitive Engineering Laboratory (CEL) at the University of Toronto (U of T) is located in the Department of Mechanical & Industrial Engineering, and is one of three laboratories that comprise the U of T Human Factors Research Group. CEL began in 1992 and is primarily concerned with conducting basic and applied research on how to introduce information technology into complex work environments, with a particular emphasis on power plant control rooms. Professor Vicente's areas of expertise include advanced interface design principles, the study of expertise, and cognitive work analysis. Thus, the general mission of CEL is to conduct principled investigations of the impact of information technology on human work so as to develop research findings that are both relevant and useful to industries in which such issues arise.

Current CEL Research Topics

CEL has been funded by Atomic Energy Control Board of Canada, AECL Research, Alias|Wavefront, Asea Brown Boveri Corporate Research - Heidelberg, Defense and Civil Institute for Environmental Medicine, Honeywell Technology Center, Japan Atomic Energy Research Institute, Natural Sciences and Engineering Research Council of Canada, Rotoflex International, and Westinghouse Science & Technology Center. CEL also has collaborations and close contacts with the Mitsubishi Heavy Industries and Toshiba Nuclear Energy Laboratory. Recent CEL projects include:

- Studying the interaction between interface design and adaptation in process control systems.
- Understanding control strategy differences between people of various levels of expertise within the context of process control systems.
- Developing safer and more efficient interfaces for computer-based medical devices.
- Designing novel computer interfaces to display the status of aircraft engineering systems.
- Developing and evaluating advanced user interfaces (in particular, transparent UI tools) for 3-D modelling, animation and painting systems.

CEL Technical Reports

For more information about CEL, CEL technical reports, or graduate school at the University of Toronto, please contact Dr. Kim J. Vicente at the address printed on the front of this technical report.

The Design and Evaluation of Transparent User Interfaces: From Theory to Practice

Beverly L. Harrison

A thesis submitted in conformity with the requirements
of the Degree of Doctor of Philosophy
Graduate Department of Mechanical and Industrial Engineering
(formerly the Department of Industrial Engineering)
University of Toronto

Abstract

The central research issue addressed by this dissertation is how we can design systems where information on user interface tools is overlaid on tbe work product being developed with these tools. The interface tools typically appear in the display foreground while the data or work space being manipulated typically appear in the perceptual background. This represents a trade-off in focused foreground attention versus focused background attention. By better supporting human attention we hope to improve the fluency of work, where fluency is reflected in a more seamless integration between task goals, user interface tool manipulations to achieve these goals, and feedback from the data or work space being manipulated.

This research specifically focuses on the design and evaluation of transparent user interface "layers" applied to graphical user interfaces. By allowing users to see *through* windows, menus, and tool palettes appearing in the perceptual foreground, an improved awareness of the underlying workspace and preservation of context are possible. However, transparent overlapping objects introduce visual interference which may degrade task performance, through reduced legibility. This dissertation explores a new interface technique (i.e., transparent layering) and, more importantly, undertakes a deeper investigation into the underlying issues that have implications for the design and use of this new techique.

We have conducted a series of experiments, progressively more representative of the complex stimuli from real task domains. This enables us to systematically evaluate a variety of transparent user interfaces, while remaining confident of the applicability of the results to actual task contexts. We also describe prototypes and a case study evaluation of a working system using transparency based on our design parameters and experimental findings.

Our findings indicate that similarity in both image color and in image content affect the levels of visual interference. Solid imagery in either the user interface tools (e.g., icons) or in the work space content (e.g., video, rendered models) are highly interference resistant and work well up to 75% transparent (i.e., 25% of foreground image and 75% of background content). Text and wire frame images (or line drawings) perform equally poorly but are highly usable up to 50% transparent, with no apparent performance penalty. Introducing contrasting outlining techniques improves the usability of transparent text menu interfaces up to 90% trasnparency. These results suggest that transparency is a usable and promising interface alternative. We suggest several methods of overcoming today's technical challenges in order to integrate transparency into existing applications.

Acknowledgments

Funding for this work and for the research laboratory was provided by the Natural Sciences and Engineering Research Council of Canada, Alias|Wavefront and Silicon Graphics Inc., Apple Computer, the Information Technology Research Centre (ITRC), Xerox PARC, and the Ontario Telepresence Project. I would like to especially thank Dr. Hiroshi Ishii previously of the Human Interface Laboratory at NTT Japan, now at the MIT Media Lab, for his significant input. Working with him during his sabbatical in Toronto significantly helped shape my thinking about not only this research but also about many creative aspects of design.

My supervisors, Professors Bill Buxton and Kim Vicente, provided invaluable support, insights, and creative freedom. Working with them was, and continues to be, a pleasure. I hope this dissertation and the corresponding publications do their efforts justice and reflect the quality they helped inject into this work. What I have learned from them I carry with me.

This dissertation was also greatly improved by the comments and feedback from my committee: Professors Marilyn Mantei, Mark Chignell, Saul Greenberg, and Joe Paradi. I am grateful for the their time and efforts in improving my work.

Comments, suggestions, and enlightenment along the way are also gratefully acknowledged from Professor Colin MacLeod, Professor Chris Wickens, Dr. Shumin Zhai, Dr. Gordon Kurtenbach, Dr. Thomas Baudel, and members of the Input Research Group, the Graphics Lab, and Cognitive Engineering Laboratory (who generously often acted as experimental subjects). The staff at Alias|Wavefront were an invaluable resource as I worked through the technical complexities and they were quite simply a wonderful group of creative talent to work with. I hope the quality of this work does justice to these people.

TABLE OF CONTENTS

1.0 Introduction

Years of diligent experimentation, design, and testing have gone into creating computer systems which are more "user friendly". One of the most notable events was the release of the Alto personal computer (Thacker et al., 1979; Thacker, 1986) and subsequently the Xerox Star desktop metaphor and graphical user interface in 1981 (Smith et al., 1982; Johnson et al., 1989), which soon became widely known as the Apple Lisa desktop after its commercial release in 1984. These widely known examples clearly drew on the seminal work of Ivan Sutherland (1963, MIT Lincoln Lab) and Douglas Englebart (1968, re-capped in 1994) which demonstrated non-command line based interactions with iconic objects, a computer mouse, and hypertext objects. These fundamental developments changed the interaction between humans and computers to more natural physical object–based metaphors. Two key concepts were created to support the transition from physical world skills to computer–based skills: the graphical user interface (e.g., Xerox Star) and direct manipulation interaction (Shneiderman, 1982), which allowed users to act directly on the object of interest within the system.

Computer desktops were designed to mimic physical desktops, including specific functional objects, such as the trash can. Through the use of metaphor and virtual objects, people's work habits, routines, and everyday skills could be more readily supported by automated tools, which presumably would be easier to learn. With the invention of a virtual desktop came the concept of direct manipulation, where operations are performed on the appearance of an object itself. This was thought to provide a more natural means of issuing commands to the computer, which would mimic current work practices (e.g., picking up a folder, moving it, opening it). However, the virtual desktop metaphor breaks down in a number of respects, since it cannot incorporate a number of physical properties of real desktops. The size of the work space is limited by the size of the screen, which is clearly a much smaller area than the physical desktop. As a result, more layering and overlapping are incorporated, objects are smaller and re-sizable to fit the space, and some techniques unheard of in the physical world had to be invented for manipulating objects and tasks (e.g., mouse clicks, highlighting, and cursors). New mechanisms were created to simulate the functions provided by physical tools such as pencils, erasers, rulers and storage containers such as file drawers, cabinets, or stacks. Obtaining and changing virtual tools is still somewhat analogous to opening your pencil drawer, except that the virtual pencil drawer often looks more like a painter's palette and the tools are small "pictorial pots" within it. The virtual palette typically sits on top of the user's drawing or document but it can be moved aside to facilitate working on a particular area (Figure 1 and 2).

FIGURE 1. Regular Tools and Desktop

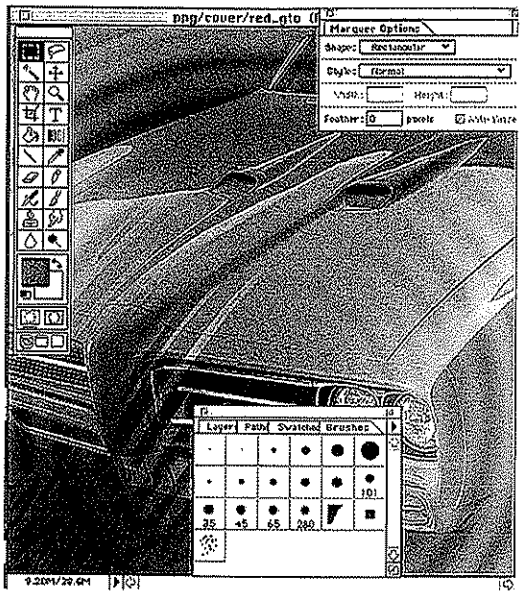


FIGURE 2. Virtual Tools and Virtual Desktop

While advances in technology have undoubtedly improved the ways in which people interact with computers and the time it takes to learn how to work on computers, there are still many opportunities for further improvement. One consequence of the virtual work space size limitation is the increase in layered, overlapping tools. These overlapping objects or items obscure the underlying information. In most computer applications, the underlying area is the document, drawing, spreadsheet, or other large data area, which is the heart of the user's work and is hence of primary interest and importance. The top "layer" contains the data manipulation tools or dialog tools for communication with the computer. To maintain a sense of context and of the state their work is in, users clearly cannot have too many tools visible and available at any one time (thereby obstructing the work). This reflects the trade-off between instantaneous access to tools versus visual access to the user's work in progress, given the limited amount of display space available. Users must often use and then "put away" tools in order to keep a view of their data or work area. Putting away and taking out tools usually means closing and opening virtual tool palettes or virtual windows, thereby increasing both time and effort.

There are a number of interesting technical enhancements which can potentially improve the fluency with which users switch tools and manipulate their work. Properly designed tools and user interaction techniques mean that accessing and applying tools should be straightforward and the resulting effects on the user's work should be immediately observable. The switches between manipulating the work or data and manipulating the tools should require minimal effort, time, and disruption. Improved fluency is reflected in a more seamless integration between task space and tool space and the transitions between these. This seamlessness can be divided into three sub

task categories: the ability to efficiently find and select the desired tool at the time it is needed, the ability to use that tool to affect the work and see the results, and the ability to easily move between searching for tools and actively working. In the context of our research, we interpret these factors as: the ability to focus visual attention on the tools (or foreground layer) and hence identify and select a tool, the ability to focus visual attention on the work or data (typically the background layer), and the cost or effort of switching between the two layers. Note that we refer to foreground and background as a means of differentiating the layering of perceptual or pseudo-physical space, though these terms can also delineate a cognitive task space of active and passive sub-tasks.

1.1 Thesis Statement

This dissertation provides an in-depth empirical investigation of transparency user interfaces. The intent is to provide experimentally derived, practical guidelines for interface designers interested in integrating transparency into their systems. To this end, we describe factors affecting the choice of level of transparency, selection response time, and selection accuracy. We illustrate a method for testing possible design combinations. Finally, we discuss some of the technical constraints and issues which must be taken into consideration if such technologies are to be implemented, particularly in transferring this technology to commercial applications. We make recommendations for implementing transparent user interfaces that can provide a variety of effective and efficient transparent tools.

We put forward the notion that transparency is a useful interface mechanism which improves awareness of multiple layers of functionality, and that an understanding of transparency is pre-requisite to designing more sophisticated see-through interface tools. When properly designed, transparent tool palettes, menus, and windows can support accurate selection while still preserving a visual awareness of the underlying information.

This dissertation illustrates a means of moving from empirical research to applied problem domains, within the context of designing and evaluating transparency user interface tools.

1.2 Foreground and Background

In the Introduction section, we discussed the notion of layers, where a desktop or workspace typically appeared *under* the interface tools to manipulate that work. This layering of windows and interface tools creates a visual foreground and background determined by perceptual cues such as occlusion. Most application domains are characterized by such a layering, with the data

in the underlying plane. Our ability to visually focus on any particular item is determined by the perceptual affordances at any given moment in time (e.g., legibility, size, extent of occlusion, number of interfering visual sources). This perceptual or visual definition of foreground and background is based on the affordances of the technology.

The concept of foreground and background can also be understood as cognitive activities (e.g., Buxton, 1995). A foreground task is the currently active work context or the activities that are in the fore of human consciousness. These are intentional activities, such as, typing or reading a letter. Foreground tasks tend to be bursts of "high bandwidth" activity in terms of human attention and effort. The background task is a passive activity, taking place in the periphery, which we may, or may not, be monitoring. These activities include examples, such as, hearing the person in the next office on the telephone, or having a motion sensor turn on a light. Background or passive tasks are ones which we typically retain a peripheral awareness of. These tasks tend to be persistent but "low bandwidth" in terms of human attention and effort. Ideally, we want technology to support not only this task structure but also the transitions between these two states.

The visual and cognitive understanding of foreground and background can be complementary. Often passive tasks are moved in the visual background (either by users or by the system) so that users can better see the active task. Ideally, we wish to map users' cognitive expectations and activities to an appropriate technological design which visually supports their requirements. The creation of transparent user interfaces strives towards attaining this goal.

For the most part, this dissertation discusses foreground and background in the visual perception sense, although it should be obvious that it is possible to map cognitive task demands to this physical design. In fact, it is desirable to create designs which support a naturalistic mapping to this inherent cognitive task structure.

1.3 Key Definitions in Visual Attention

In order to support improved visual awareness of a work area, we focused on the use of superimposed transparent layers. We define *visual awareness* as the uninterrupted and non-occluded visual perception of an image or object. Visual occlusion occurs whenever one object appears over another, blocking out all or portions of the underlying object. There are varying degrees of visual occlusion or visual awareness, which depend upon the extent to which the image is blocked. People are very adept at compensating for missing portions of an image, provided that enough of the image is still visible and that the obscured object is a familiar one.

This is often referred to as *gestalt* perception, or perception of the whole (Kohler, 1929). One way that we can vary the extent of visual awareness of obscured objects is by changing the degree of transparency of the foreground item. In effect, these differences in awareness are comparable to looking out of a clear glass window, a slightly frosted window, and a highly frosted or textured glass window. The object in the background is completely, somewhat, or only slightly visible. Seeing these background objects may result in some corresponding decrease in the ability to discern the features and attributes belonging to the foreground object, particularly if there are few depth cues. Unlike the window analogy, computer systems which incorporate layers typically do not have stereoscopic or 3-D depth cues; any perceived layers are a result of interpreting 2-D cues, such as occlusion, linear perspective, size constancy, or texture gradients. However, transparency itself provides a subtle depth cue, in addition to the more traditional 2-D depth cues. Objects lying behind a partially transparent window are necessarily blurred, while objects in front of this window are clearly perceived. This allows us to make judgments about what is nearer or further away (e.g., Zhai, Buxton, and Milgram, 1994). Within a virtual windowing system where objects are not actually at differing depths, this transparency feature makes them appear in perceptually distinct layers. This influences the extent of our visual awareness of multiple competing sources of information and our ability to separate out which features belong to objects in a particular layer. Visual awareness is a combination of both external physical constraints providing sufficient visual cues, and of internal human processing capabilities, such as attention and effort, being exercised to note the visual phenomena of interest.

Whenever there are multiple objects in a scene, we must make choices about what to attend to and when. We select the precise object to focus on at a given moment in time, typically based on the task context and moment–to–moment task goals. We see this object in detail and deliberately ignore all others. This characterizes focused visual attention (or *focused attention*). The other critical component is that of *divided attention*. There is still debate about whether people can simultaneously visually monitor two or more items at once (multi–channel or parallel processing) (Wickens, 1984), or whether they merely monitor a single item at a time and rapidly switch back and forth between items to divide attention (single channel, capacity model) (Moray, 1967; Kahneman, 1973). In either model of divided attention, visual proximity of the information sources and switching cost are critical. (We discuss the differing theoretical models of attention, object perception, and the implications for our research in more detail in Chapter 3). Divided attention supports both the cognitive model of multiple, competing active tasks and a model of active-passive task switching.

In the case of virtual work areas with overlapping user interface tools, accurate item recognition and selection is a fundamental requirement. For our research, since we were primarily interested in scenarios containing these visually overlapping items which must be moved out of the way, we focused on attentional models which include a *switching time and cost* (i.e., capacity models such as those proposed by Moray, 1967, 1969). Part of this time and cost is associated with a change in focus of attention, while another part is associated with the window management necessary to bring items into view such that item identification and selection is possible. Transparency affects both the physical effort required to bring items into view (or whether they are already in view) and the mental effort needed to change the focus of attention back and forth between multiple objects. Finally, when choosing between two sources of information, the extent to which these sources differ or are similar greatly influences the *visual interference*. In particular, if we have overlapping transparent objects, we anticipate that the information on one layer will affect the extent to which we can determine what appears on the underlying layer. For example, layers such as text over text are likely to be more difficult to separate than text over video; layers of matching colored items will show higher visual interference than dissimilar colors. We anticipated that visual interference would play a key role in both focused and divided attention performance with transparent layers.

1.4 Task Description

One potential advantage of transparent interfaces is the preservation of context or inherent task structure. There are many examples of task scenarios where users work with two windows, two displays, or two versions of documents. One task is inherently active while the other is passive (or a peripheral activity). We may wish to preserve a sense of awareness of the background task while focusing on the foreground task. Our primary data or work area may lie within one window or view while the data manipulation tools appear in other windows or palettes. Figure 3 shows a snapshot in time of an actual working session for an artist doing image editing and image touch ups using a popular commercial application program. At any given moment, several windows and palettes are visible, which allow the artist to quickly change his airbrush and paint brush attributes. The artist must then move or close these tools in order to continue work on the car image. In a typical session, the artist has 3–4 windows open simultaneously. Also, notice that the typical working style is not to overlap any interface tools windows. The artist attempts to maximize the speed with which he can access tools and options by working with two perceptual layers, as often as possible. However, what he really wants is a work environment with no windows or icon palettes at all so that he may work more continuously on his image.

Figure 4 shows another typical scenario in a commercial painting application, with transparency added, where help screens are momentarily active while we read them; the context they are called from temporarily becomes a background task. In this case, the help screen shown was taken from the product we later evaluated as a case study. Again, this was a commercial painting and drawing application (though not the same one as that shown in Figure 3).

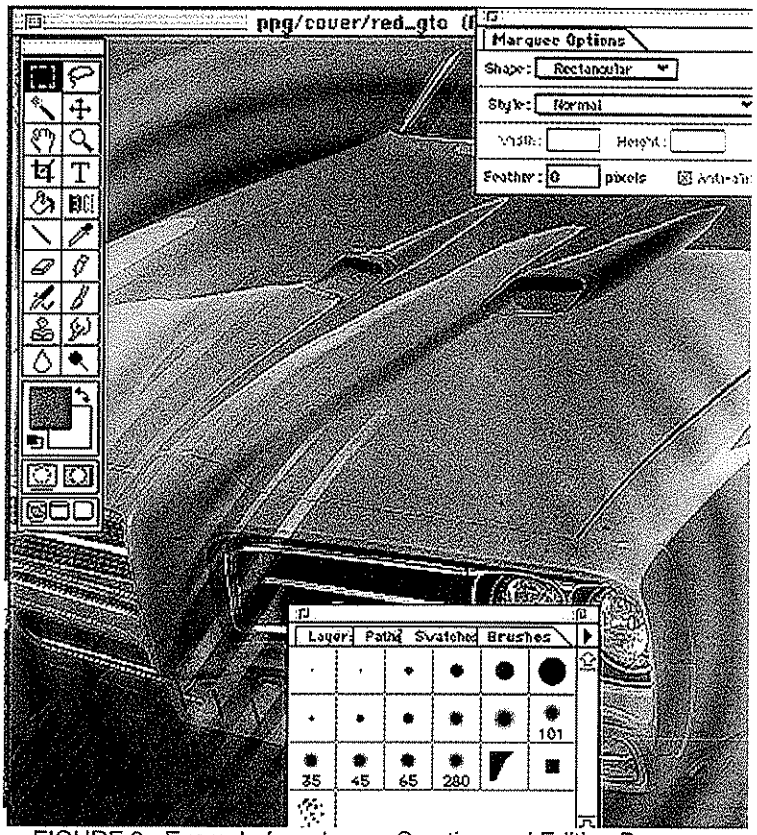


FIGURE 3. Example from Image Creation and Editing Program

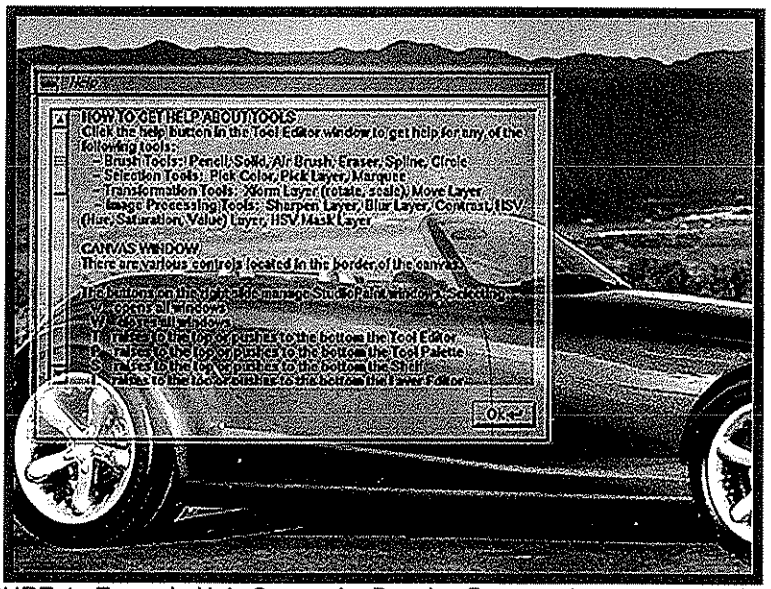


FIGURE 4. Example Help Screen for Drawing Program (transparency added)

One method of assessing the usage patterns for layers of interface tools and the extent to which they interfere with a primary task is to review video taped work sessions. We not only video taped users working with a standard opaque drawing and painting application, but we also reviewed the tutorial videos which were used for customer training. By playing these videos in fast forward mode, we were able to condense lengthy sessions into much shorter time spans where the window management activity was still clearly visible. From these reviews, we observed 3 or 4 windows open consecutively with users flipping rapidly between the foreground UI windows and the background image (work space). We estimated that more than 50% of the users' actions were to resize, move, open and close UI tools, while the remaining actions were drawing and painting strokes or modifications directly on the work (i.e., real work). Note that this was not intended to be an exhaustive study of many users, controlling for task and individual differences. Rather, we wanted to globally assess the extent of potential benefit transparency might have in this particular target application. Secondly, we also wanted to see if we could find methods of objectively quantifying work fluency or disruption to work attributed to layered interfaces. We believe this can be measured in part by determining window management activity in terms of both number of key strokes and total time versus the time spent actively executing elements of the task. Users also have clear subjective feelings about how productive and/or frustrating given work sessions are. We will discuss our methods of assessing transparency in more detail in Chapter 7 when we review the product case study. However, our primary goal is to understand the design parameters such that transparent windows are a workable and useful solution, independent of the more precise measures of work fluency. Ideally, work fluency would be positively impacted, but this requires longitudinal studies in addition to the measures we have proposed above. As such, these long term usage studies are not within the scope of this dissertation. The design issues related to creating transparent user interfaces and minimizing visual interference are addressed in detail in the experimental results and in the case study presented in Chapter 7.

The results of our video review of work sessions gave us reason to believe that our target application of painting and drawing would be a good case study for transparent windows. In addition to this particular application, there are many other domains which are characterized by multiple layers, one of which is the work space while the other layer(s) are the manipulation tools or a passive background task.

In collaborative work, we often have a shared document or drawing (foreground) that we are co-authoring and a video connection or window to our colleague (for awareness – background). We wish to preserve an on-going sense of "connected-ness" to our collaborator while actively working on the text in the document. This was obviously one of the goals illustrated by the

systems shown in Figures 12, 13, and 14 in Chapter 3, which reflect the integration of task space and person space for collaborative appplications, using transparent overlays.

In addition to the simplistic help scenario shown above, we are also interested in using transparency as a means of providing more explicit "show and tell" help. For example, in the TeamWorkStation system described in Chapter 3, in one scenario semi-transparent images of an expert's hands were shown superimposed on a schematic drawing. The expert explains aspects of the drawing using pointing and gestures to specific items. Clearly, this illustrates the tight potential coupling transparency can provide between a help system and the context from which it is called. Transparency may allow us to better preserve these relationships in many other help system contexts.

A number of other situations arising out of our day-to-day work are outlined below, reflecting the diverse range of possible applications.

- working on a document when a dialog box or warning message interrupts
- a pull-down menu (or pie menu) which may temporarily block part of our current window (The selected menus items may go on to create further dialog boxes of their own.)
- using tear-off tool palettes (which behave as tiny overlapping windows)
- viewing a live video conversation with one person while monitoring several Portholes-like, or passive, connections to others for peripheral awareness of their availability
- using an interactive dialog box to change the drawing or modeling characteristics of an underlying image, model, or animation

In addition to those tasks listed above, obviously the systems described in Chapter 3 suggest other promising applications and interaction techniques which rely on transparent layers and minimizing visual interference between these layers. By their very nature, many of these proposed task pairs have an implicit active and passive task. We need a *peripheral awareness* of the passive task while we temporarily divert most of our attention to the active task. The extent of this awareness determines the extent to which we must divide or focus our attention. We also must consider the visual contents and distinctiveness of the two layers within the task. How similar are they? What is the information density and level of detail of each? This determines how much interference may result when we focus our attention on one object. These characteristics may be unique for each task. In our particular target task the background layer is a drawing or painting (i.e., a solid image) and the foreground layer is a combination of text, graphical user interface items (e.g., sliders, buttons, color wheels), and window management objects (e.g., title bar, borders, close box, resize). However, we are interested in what happens if

users also choose to include text or line art figures in their drawings. We need a clear understanding of the interaction between transparency level, legibility, and interference. To this end, we have conducted a series of progressively more representative experiments described in Chapters 4 and 5, based on foundations from the psychology literature summarized in Chapter 3.

1.5 Research Approach

This research spans a continuum, moving from field observations to theoretical laboratory experiments to applied evaluations and field studies (Figure 5). In Chapter 1 (Introduction), we outline the goals of the dissertation, some basic conceptual definitions, and a description of the task environment under consideration. This task descriptions outlines the practical problems observed in the field and is later used to guide our experimentation and subsequent case study.

Chapter 2 structures the task-based problem (identified in Chapter 1) into a framework for research. A three dimensional taxonomy is created to help categorize technological solutions and position current research systems. A conceptual framework is proposed which facilitated structuring the subsequent experimental work.

The issues under consideration are then set within the context of prior theoretical and applied work (Chapter 3). We draw extensively on the visual attention literature and its potential application to transparent user interfaces. We additionally provide examples of existing systems which make use of transparency as a further motivation for the novelty and applicability of this research.

To obtain a preliminary estimate of the transparency level that best supports human attention, we have conducted formal experimental studies with controlled models and simulations (Chapter 4). These experiments were based on theoretical models of attention and visual interference, using the Stroop Effect and relying on existing literature. We anticipated that these results would provide us with threshold values of the worst case scenario (i.e., highly interference sensitive task).

Subsequently, we introduced more representative task elements into the experimental scenarios and empirically evaluated these (Chapter 5). These combined experimental results provide us with quantitative data on how well users can focus attention on either the foreground or the background layer and how high the interference is between the two layers by varying degrees of semi-transparency. The experiments measure subjects performing item identification and selection tasks, reflecting real world application demands.

However, we realize that controlled experiments address a restricted set of design dimensions since a finite (small) number of independent variables are manipulated, and a finite (small) number of dependent variables are measured. Our experiments initially used simplistic stimuli to allow us more robust measures of interference using well developed, pre–existing methods (for example, the Stroop Effect). Real commercial applications consist of a much richer design space and more complex "performance" goals. Therefore, we have developed several prototype systems which are more representative of these commercial applications (Chapter 6) (again based on the selected task domain described in Chapter 1). We evaluated these prototype systems and observed user behavior to gain further insights into the design of transparent user interfaces.

Finally, we modified a working commercial application to include transparent user interfaces (Chapter 7). A case study of this product was carried out. This combined research program allows us to further formulate research issues while maintaining external (real-world) validity. The empirical and prototyping approaches were partially conducted in parallel and as iterative design evolution. Thus, the research described in this dissertation documents a complementary combination of both controlled empirical studies and methodology with more representative applied research. Below we present a "road map" of this dissertation (Figure 5) which reflects the progression of work.

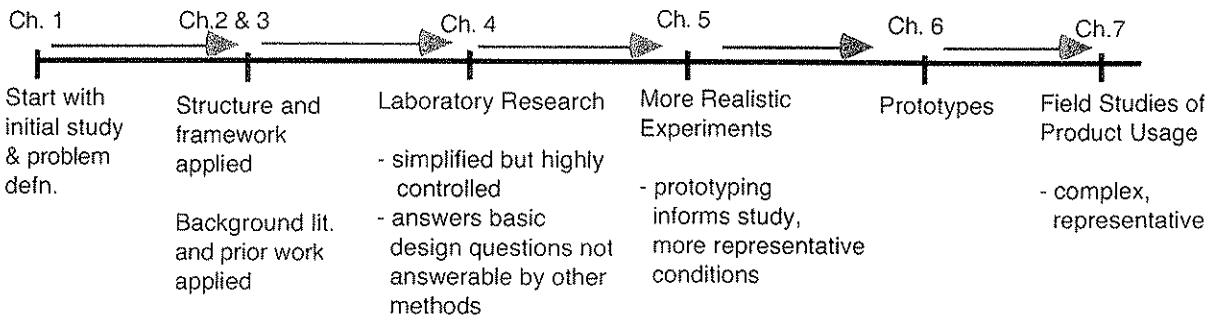


FIGURE 5. Structure of the Dissertation.

2.0 Structuring the Problem[1]

In this chapter we propose a framework which allows us to formulate our research questions, assess existing systems and prior work, and characterize new systems and emerging bodies of research as they relate to transparent layered user interfaces. We examine the design and evaluation issues as they relate to the technology, psychological attributes of human attentional performance, and task demands.

2.1 Technological Framework

The small amount of display real estate available relative to the amount of data to be displayed presents a real challenge to user interface design. To date, two main strategies have been applied to the problem. We have classified these strategies as: 2-D space multiplexing (user interface items are accessed by spatial location) and time multiplexing (interface items are accessed in sequence temporally). We propose a third strategy: 3-D depth multiplexing (interface items are transparently overlaid). (As used here, we define user interface items to include a variety of graphical objects including menus, windows, icons, and palettes.) These strategies are defined in terms of the implications that the particular design has on information manipulation. In particular, the user can accurately focus on one source of information at a given instant in time. However, the goal is to partition attention among multiple sources of information through explicit or implicit attentional switching. The design strategies directly influence the mechanisms for making available these alternating sources of information. Does the user switch attention by a spatial change in focus (e.g., a driver focusing on the speedometer and then the road)? Is an explicit switch required which presents the alternate item in the same spatial location but only after a user–initiated request and a slight time delay (e.g., flipping pages in a book)? Is switching some combination of these?

In the first strategy, space multiplexing, the screen is partitioned, or *tiled,* into a number of non-overlapping windows (Figure 6). The advantage of this approach is that users can clearly see and identify each item. In order to switch attention from one item to another, users need only change their focus of attention spatially. We call this strategy, the *space multiplexed* strategy. The

---

1 Portions of this chapter were published as Harrison, B. L., Ishii, H., Vicente, K. J., Buxton, B. (1995). Transparent Layered Interfaces: An Evaluation of a Display Design Space to Enhance Focused and Divided Attention. In *Proceedings of CHI'95,* May 6-12, 1995. Denver, CO. New York: ACM Press. 317-324.

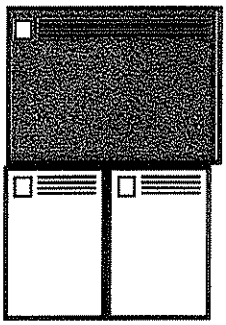


FIGURE 6. Space Multiplexed Strategy

disadvantage of this strategy is that a limited number of objects can be shown at any given time, leaving many objects temporarily "hidden". These hidden or obscured objects must be explicitly found and brought to the foreground, replacing (in spatial location) some existing item. Also the relationships or context between objects is difficult to represent by simple rules of adjacency.

In the second strategy, items lie on top of one another (*overlapped*). Only the top item is completely visible at any given time, but a mechanism is provided to rapidly change which object is visible. This is typically some combination of mouse clicking, selection, and dragging. At any one moment in time, a subset of items are visible. We call this strategy, the *time multiplexed* strategy (Figure 7a). This strategy has the advantage of representing many more objects but at the expense of increased "attentional switching costs" and a decrease in awareness of which objects are present (some are partially or completely occluded). Most frequently, a hybrid of the two strategies is used (Figure 7b).

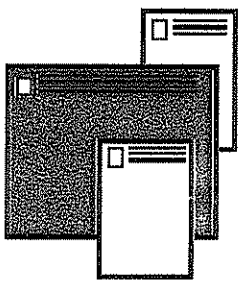
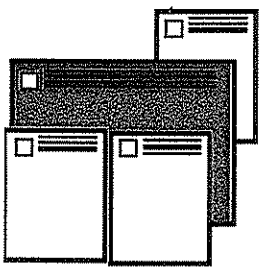


FIGURE 7. (a) Time Multiplexed Strategy



(b) Hybrid Space & Time Multiplexed

What we propose in this dissertation, however, is a third strategy. Through the use of semi-transparency in the foreground items or display, the contents of the objects underneath are visible, but with lower contrast (Figure 8) (i.e., no occlusion). Transparent items can still be tiled or overlapped. When overlapped the user can see through the top-most object to preserve a sense of awareness (and a sense of context) for the object in the underlying layer. Awareness occurs when it is possible to peripherally monitor (at minimum) what is occurring in the underlying layer. This "new" strategy we refer to as *depth multiplexing*. It can also form a hybrid interface of opaque and transparent objects, combined with tiled and overlapped display designs.

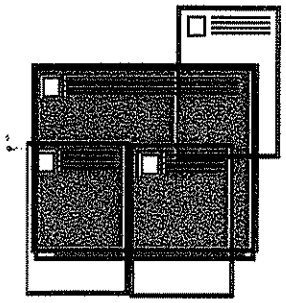


FIGURE 8. Depth Multiplexed Strategy

We wish to classify and evaluate a variety of semi-transparent interface items. Broadly defined, these include menus (pull-down, pop-up, and radial or "pie" menus), palettes (tear off tool menus), dialogue boxes (especially interactive scrolling dialogues), windows, and help system screens. These items appear in many applications and, at least temporarily, obscure part of our work surface. The degree to which they persist (seconds versus minutes or hours) largely determines how disruptive they may be. In many situations, our primary task or the area of interest becomes the background layer while these objects appear in the foreground to enable us to carry out activities that are ultimately reflected in the now obscured background layer. Transparent interfaces allow us to observe these changes without obscuring the task layer. We do not need explicit window movement or manipulation strategies to switch attention between the menu or dialogue box and the underlying item of interest (which may be providing visual feedback based on our actions). For example, we may change font size, perform cut–and–paste operations, change color of items, alter size of graphical drawing objects, or manipulate other attributes.

On the one hand, the depth multiplexing approach offers the best of both worlds: windows need not be tiled to be visible. Hence, ideally, less information is obscured. On the other hand, the potential for the content of one window interfering with another above or below it is introduced. When using transparency without the benefit of 3-D depth cues, our visual system blends the combined images together. The extent to which we are able to accurately separate information out into the correct layers determines the extent of visual interference. This depth multiplexing strategy raises questions about legibility, visual interference and what types of information this technique is best suited for. The objective of our research agenda is to develop a more formal understanding of the constraints of such an approach.

2.5 Psychological Framework

We are concerned with three critical attentional components: a person's ability to focus attention on a foreground item or a background item, a person's ability to separate the visual characteristics of each source and focus on any single item with minimal interference from other items, and the switching cost (time, mechanism, learning, awareness) of shifting attention from one item to another. For the sake of explanation, we refer to the topmost item or objects as being in the foreground and the overlapped or obscured items as being in the background. This analogy matches depth cues from visual perception although, in fact, the items we are presenting are not

separated by any visual cues that indicate 3-D depth. Partial occlusion is a well known depth cue in 2-D imagery or scenes and transparency, in particular, has been shown to be a strong 3-D depth cue (Zhai, Buxton, and Milgram, 1994). As we previously mentioned, tasks which are in the fore of human consciousness typically map to the visual foreground, while passive or peripheral tasks map to the perceptual background.

To facilitate focused attention (e.g., ignoring information from the background layer while focusing on the foreground), we want to make the attributes of the information on foreground objects as different from the background as possible. (Highly dissimilar information causes less visual interference and is more readily separated into layers upon which we can subsequently focus.) We also wish to reduce the visibility of the background objects to minimize interference. Opaque foreground objects are clearly the most highly visible and have the least potential interference from the background content. The reverse is true for tasks that require focused attention on the background layer while the foreground layer is simultaneously visible. In this case, higher levels of transparency will improve legibility of the background. Finally, for divided attention (being able to see both foreground and background layers), we need to support simultaneous visibility of both layers. However, the user must still be able to separate which features belong to the foreground and which features belong to the background in order to accurately perceive the objects. There are many ways of achieving differentiation between layers (with varying success), such as different colors, content attributes – analog (images or graphics) versus verbal (text based), font sizes or styles, etc. Our focus is on manipulating features of the system (i.e., the transparency of the layers) and the user's ability to alternatively focus on the foreground or the background.

Clearly there is a trade-off between foreground and background focused attention. We need to support this trade-off since most real world jobs require *both* focused and divided attention. We have characterized the trade-off in Figure 9 which provides an attentional framework for this research. We have used level of transparency as the visibility control variable. From this analysis, we can project that the preferred degree of transparency is determined by the trade-off of supporting both focused attention for the foreground and focused attention for the background. Presumably the intersection of these two performance curves reflects the optimal divided or time sharing attention point. As degree of transparency increases, it gets easier to focus on the background but more difficult to focus attention on the foreground object since visual interference is increasing. The optimal transparency (OT) is a result of a trade-off. The curves and the location of optimal transparency in the figure are hypothetical but may reveal the trend. The non-linear nature of the curves is based on standard functions used in psychophysics research. Finally, we believe that the proposed curves will mimic in shape those curves observed

in typical psychophysical and perceptual experiments where there are lower and upper performance thresholds.

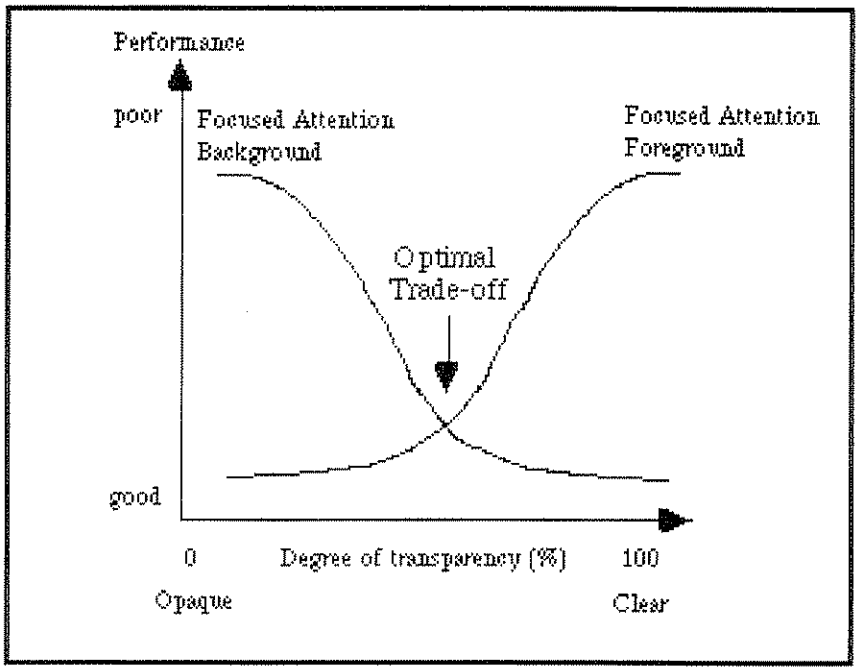


FIGURE 9. A Hypothetical Model of Transparency Selection.

2.2 Display Design Framework

We propose a design space that captures the above three multiplexing strategies (space, time and depth) and applies, in general, to foreground and background interface layers (Figure 10 and Figure 11). This design space allows us to methodically categorize and investigate both existing technologies and more novel technologies. We briefly summarize how certain systems are positioned within this design space. More detailed descriptions of these systems are provided in the next chapter.

In one dimension (upon which this dissertation focuses), we vary the level of transparency/opacity between the two displays. *Fully opaque objects* reflect traditional window, palette, and menu design in current graphical user interfaces. *Fully transparent designs* reflect some of the more advanced interfaces such as those used in Heads Up Displays (HUDs) in aviation (e.g., Larish and Wickens, 1991; Wickens, Martin-Emerson, and Larish, 1993) or in the ClearBoard system (Ishii and Kobayashi, 1992; Ishii, Kobayashi, and Grudin, 1993). In HUD design, aircraft instrumentation (a graphical computer interface) is superimposed on the external real world scene using specially engineered windshields (see Figure 21). In the ClearBoard work, a large drawing surface is overlaid on a video image of the user's collaborative partner (see Figure 13). *Semi-transparent designs* include such things as video overlays commonly used in

television (like those used in presenting sports scores while the game is playing), "3-D silk cursors" (Zhai et al., 1994), or ToolGlass–like tool palettes (Bier, Stone, Pier, Buxton, and DeRose, 1993; Bier, Stone, Fishkin, Buxton, and Baudel, 1994; Kabbash, Buxton, and Sellen, 1994; Kabbash and Buxton, 1995). "Silk cursors" are semi–transparent, volume–based cubes which act as volume cursors in 3-D virtual spaces. The volume is used to "enclose" an item in 3–space to select it. The transparent surfaces of the volume allow users to determine how far into or outside of the volume the target item lies. ToolGlass palettes are completely transparent "stencils" which users align to an object and then click through to apply the selected function.

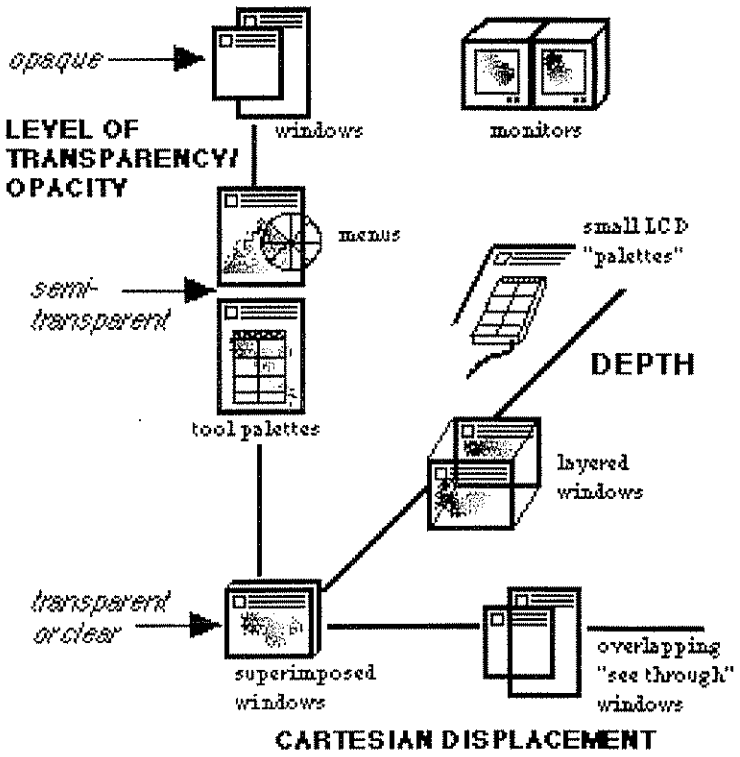


FIGURE 10. Design Space Dimensions

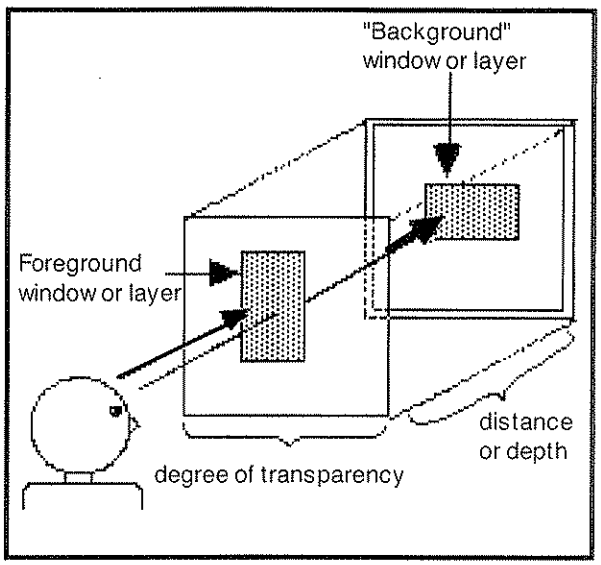


FIGURE 11. Concept of Layered Displays

Along another dimension, we can vary the *perceived depth* of the planes between two displays, where one image appears closer to the user while the other is in the background. This can be accomplished using half-silvered mirrors, polarizing filters, or special transparent LCD displays (creating binocular disparity or stereopsis). In this case, the user looks through the display presented in the foreground to see the display presented in the background (e.g., Kobayashi and Ishii, 1994). Layers on this axis are distinguished by both transparency and depth. There are limited examples of such systems. Knowlton (1977) used graphical overlays projected downwards onto half-silvered mirrors over blank keyboard keys to dynamically re-label buttons and functions keys (e.g., for telephone operators) (see 3, Figure 18, p. 27). Schmandt (1983) built a system to allow users to manually manipulate and interact with objects in a 3-D computer space using a 3-D wand. Again a half-silvered mirror was used to project the computer space over the user's hand(s) and the input device (see Chapter 3, Figure 19, p. 28). Disney has also developed a product called the "ImaginEasel" for animators and artists. ImaginEasel keeps the user's hand and input device in the workspace (using mirrors) (see Chapter 3, Figure 20, p.28). Note that the depth and displacement dimensions are presented here for completeness but will not be examined in this dissertation.

2.3 Summary

In this chapter we described a categorization framework for the design of displays (space multiplexed, time multiplexed, depth multiplexed, hybrid designs) and we proposed a design space for classifying technologies (Figure 10) based on Cartesian displacement, depth displacement, and transparency level. These were used to understand both the existing and future systems and to demonstrate where our research fits in relative to this body of work. We additionally proposed a hypothetical model of attention (Figure 9) which allows us to formulate hypotheses about the nature of performance based on varying the level of transparency in a layered user interface context. While our proposed model is meant to be fairly comprehensive, the research described in this dissertation investigates the design and evaluation of transparency

within several boundaries. We have deliberately limited the scope of this problem to include two layers of information (and not more than two). We believe that a fundamental understanding of two layers and the inherent advantages and issues is essential before moving on to the complexity of multi–layer interfaces. Two–layer interfaces address many of the basic issues while still retaining relevance to usage patterns in our target application domain. Additionally, for completeness we described not only the scale of transparency but also that of depth. The research presented here focuses on the problem of transparency without confounding this issue by incorporating 3-D depth perception. Finally, we explicitly investigate the problems of focused attention in both a foreground and a background context. From this work, we can make inferences about the impact on divided attention. However, this dissertation does not include a direct empirical evaluation of a true divided attention task and as such, we recognize that there may be performance issues unique to such a situation which are not addressed in this dissertation.

3.0 Background Literature

This dissertation draws on a number of diverse areas in terms of supporting research and literature. We first describe a number of systems and technologies which have incorporated some component of transparency into their design. In particular, we summarize findings from research into Heads Up Display designs, one of the few transparency–related systems where any empirical evaluation has been conducted and reported. These systems reflect the diversity of transparency in existing systems and suggest some of the problems encountered and solved. Our understanding of divided and focused visual attention and of visual interference also plays a key role in formulating our subsequent experimental work. To this end, we summarize the most relevant literature in visual attention. In particular, we draw on the Stroop literature to better understand issues of interference in visual attention and to describe how our experiments are a novel use of the substantial existing body of work in this area.

3.1 Related Work on Other Systems

There are several systems which we briefly alluded to when we applied our proposed design space framework in Chapter 2, which are relevant to transparent display designs. These systems have not been empirically evaluated but do provide interesting anecdotal evidence that transparency has a number of useful applications. They additionally suggest future application domains beyond the target domain we are specifically using for our case study evaluation.

TeamWorkStation and ClearBoard

The TeamWorkStation system (Ishii, 1990; Ishii et al., 1993) incorporated shared workspaces with a transparent overlaid live video window to facilitate collaborative work between two people. In the first iteration of TeamWorkStation (TWS-1), the main work space consisted of a shared digital workspace window (the computer screen image) and a translucent overlaid video image of the user's desktop, including hands, gestures, and physical drawings. There were separate video windows of each of the collaborating partners. In the later instantiation of this system, called ClearFace, not only was the computer screen represented but both collaborators' video images were also superimposed transparently on the workspace area (Figure 12).

According to anecdotal evidence reported by Ishii (Ishii et al, 1993), users had little difficulty separating out and selectively viewing the facial images or the work space images.

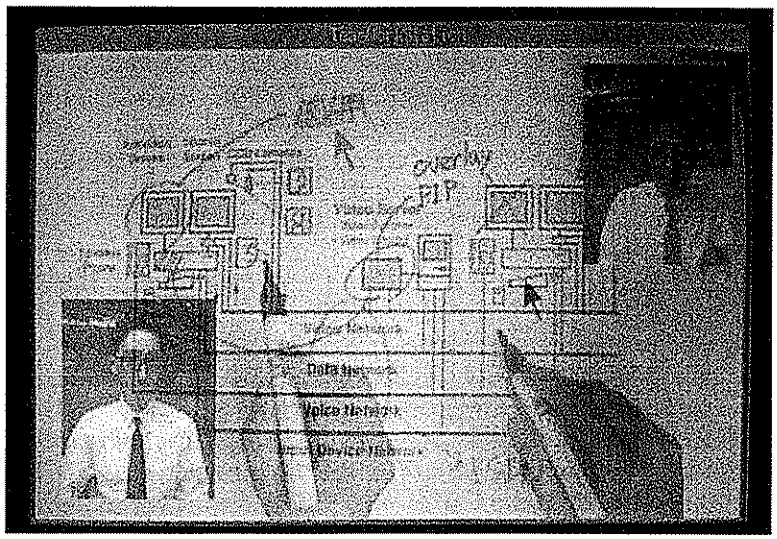


FIGURE 12. TeamWorkStation (ClearFace iteration)
(Ishii, Kobayashi, and Grudin, 1993, p. 351)

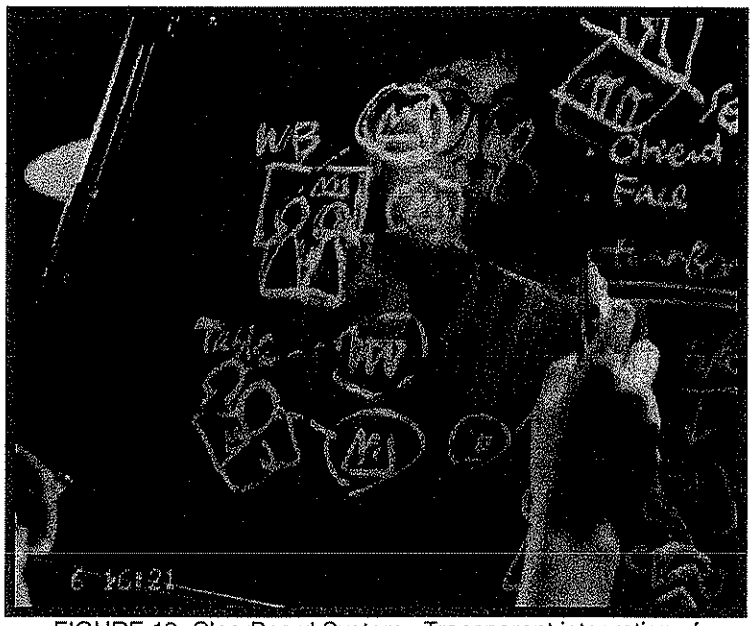


FIGURE 13. ClearBoard System. Transparent integration of
collaborative partner and video drawing surface. (Ishii and Kobayashi, 1992, p. 705)

A number of design issues noted in the TeamWorkStation system were addressed in the ClearBoard system (Ishii and Kobayashi, 1992; Ishii et al., 1993), most notably the lack of gaze direction and gaze awareness. Several simple but clever glass prototypes were built and tested to see if full-sized images would provide these clues about where a collaborator's attention was precisely focused. The premise was that this degree of awareness would smooth much of the coordination effort involved in synchronous, long distance collaboration. The final ClearBoard

system incorporated a drafting table style of drawing surface and full-sized video partners projected transparently under the drawing (Figure 13). While there were no formal experiments run on this system, strong and consistent anecdotal results indicate that the original goals were successfully met. Users could easily focus attention on either layer without interference and shifting attention between layers was achieved through a seamless shift in visual attention. Furthermore, users could easily track the direction of their partner's gaze and identify the object being looked at (including themselves).

VideoDraw

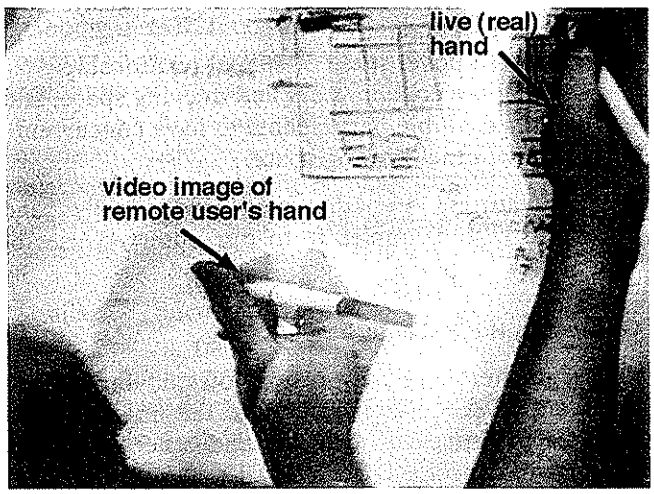


FIGURE 14. VideoDraw System. Transparent integration of collaborative partner's drawing and hands under local drawing surface. (Tang and Minneman, 1990, p.315)

Conceptually similar to the ClearBoard work, the VideoDraw system from Xerox PARC (Tang and Minneman, 1990) also integrates live video and drawing images. However, in this case it is the collaborator's hands which are shown by live video. The underlying premise was the importance of capturing gestural cues and hand movements within the context of a digital shared work space (Figure 14). In particular, VideoDraw allows collaborators to see each others' hands and each others' drawings as though they were superimposed using clear acetate. In reality VideoDraw relies on polarizing filters and overhead cameras to transmit and combine images. Collaborators draw on the surface on a recessed monitor and both their hands and the drawing marks are captured by the overhead camera and are displayed on the other person's screen. The polarizing filters ensure that video crosstalk is avoided, i.e., only the necessary remote image is transmitted. By virtue of using cameras and polarizing filters, the transparency level in this scenario is not adjustable. Images appear as 100% transparently overlaid. Studies of this prototype system were conducted to better understand the process of collaborative drawing and the role of gesture. No empirical experiments were conducted and, given the configuration of the system, technological permutations were infeasible.

Tool Glass and Magic Lens

The ToolGlass and Magic Lens work from Xerox PARC (Bier et al., 1993; Bier et al., 1994; Stone et al., 1994) represented an innovative use of transparency between two graphical user interface layers. The ToolGlass was a completely transparent (100% clear) layer between the cursor and the background work space which allowed users to apply a variety of functions in a "click-through" manner. It was roughly analogous to a clear plastic template or stencil in terms of real world tools. However, it could provide far more sophisticated functionality due to its virtual nature. Bier et al. (1993) describes ToolGlass as "a virtual sheet of transparent glass" (p. 73). Each sheet of glass contains one or more iconic tools which are aligned with an underlying object. Selecting the aligned tool by clicking on it applies that particular function to the object immediately below the tool selected. To further improve upon the physical analogy, a ToolGlass is typically moved around with a device in one hand while the individual tools are selected with the other (exactly as a template would work). ToolGlasses and Magic Lenses can be layered one on top of another, creating a multi-tool object consisting of many different clear layers and assuming all of the combined functions of the individual sheets of glass. For example, in Figure 15 we see a "gray scale" Magic Lens which removes color, a "shadow" Magic Lens which adds drop shadows, and the effect of combining both.

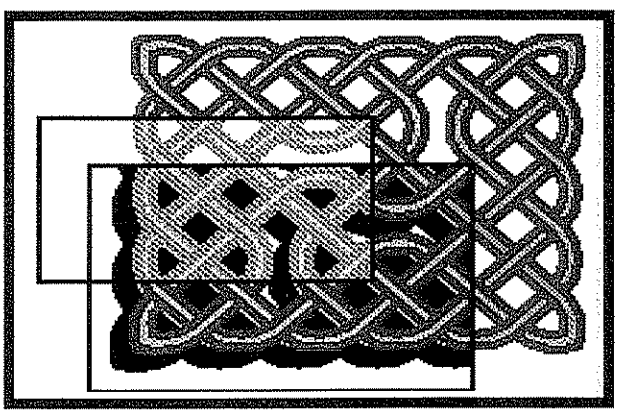


FIGURE 15. ToolGlass Example (Bier et al., 1993, p. 75)

In addition to ToolGlasses and at the same time, the concept of a Magic Lens™ was created. Magic Lenses act as special filters which can zoom in or out, alter color, filter out objects which do not match a particular color or attribute, and any number of other possible functions. As in ToolGlasses, a Magic Lens works by positioning it over the target object. Unlike ToolGlasses, Magic Lenses are analogous to physical lenses in that they are always active. Users do not need to click on them to apply their function. Magic Lenses are characterized as bordered objects which are "sort of transparent" – the object underlying them may be altered in any number of possible ways and so it is difficult to simply categorize them as transparent tools. Magic Lenses

may also be combined with other ToolGlasses or Lenses to create a multi-function tool. For example, in Figure 16 an "outline" ToolGlass item is combined with a magnifier Magic Lens. In Figure 17, we show two Magic Lenses one of which highlights roads on a map, the other highlights rivers. All irrelevant information is filtered out. These Magic Lenses visually appear to be semi-transparent.

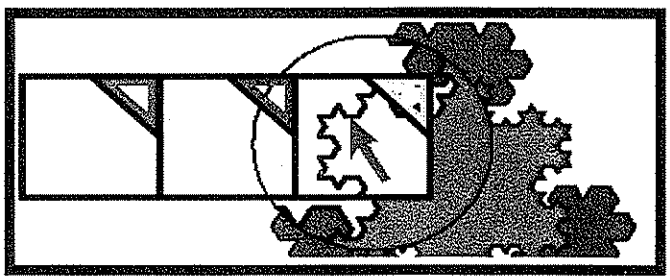


Figure 16. Sample ToolGlass and Magic Lens Combined
(Bier et al., 1993, p. 73)

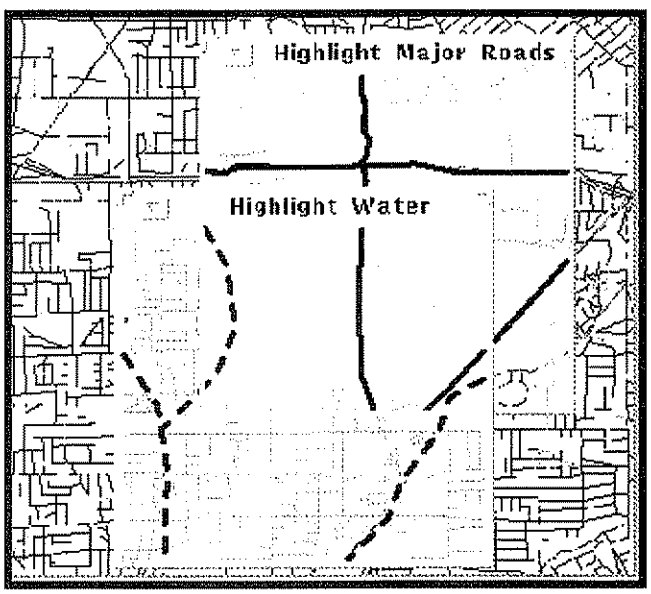


Figure 17. Two Magic Lens Filters (Bier et al, 1994, p.309)

It is clear that transparency is an essential component of the ToolGlass and Magic Lens work. Object alignment with the background layer determines the scope of the function to be applied and therefore this object must be at least partially visible. We see our research as a preliminary step in understanding what levels of transparency can be used and under which conditions transparency will work best or will not work at all (interference between the tool and the target object is too complex). Thus far, there are no empirical evaluations reported on the ToolGlass or Magic Lens work, though it is clear through anecdotal reports and trial usage that these tools

seem to work well. No control for visual interference has been reported in the literature; this would seem to be one of the critical dimensions affecting potential usage of these tools.

Translucent History in Architectural Drawings

Another interesting and unique application of transparency was implemented by Genau and Kramer (1995). In their drawing application, they represent object history by transparency. The application domain was that of architect's sketches for room layout problems where items within the room are often moved. Former locations of objects become increasingly dim as they are changed more often. The analogy here is one of layers moving further away from the user to represent object states aging in time. The closest layer (i.e., the darkest objects) is the current working set. Users may alter the ordering of layers. In this instance there is one object per layer, except for the uppermost layer which contains the current state of all objects. This approach draws on earlier work (Kramer, 1994) where translucent layers were used to record annotations on drawings (again this was described in the context of architectural drawings where the annotations were modifications to the drawing). In this case, the annotations appear to have been drawn with a lighter pen color rather than incorporating a notion of layers. However, it is clear in the later work (Genau and Kramer, 1995) that layers are an intrinsic part of the system.

While this work presents an intriguing potential application domain, no evaluation was conducted to determine either the effect of varying the levels of transparency or of the interaction effects between layers. Visual interference was not assessed and there are few users of this system (Kramer, 1994, personal communication). Transparent layers were generated by reducing the saturation of the color for the object on that particular layer. The graduation of transparency levels was not measured. Again, we see that our evaluations and design research can complement and inform the Genau and Kramer work by determining appropriate levels of transparency and by suggesting alternative implementation methods. Our evaluation may also provide insights into potential visual interference issues for this application.

Transparent Layers in the CaveDraw System

The CaveDraw system (Lu and Mantei, 1993) supports individual and synchronous group drawing. The feature of most interest is its use of transparency layers (layers are like sheets of see-through paper or acetate). Users can create, hide, or select any layer. When new layers are created (i.e., superimposed on existing layers), the work on previous layers dims to a lighter color (i.e., less saturated) to reduce potential interference. The top-most layer is the most visible. In this way, as a drawing evolves, a stack of layers is created. As layers are superimposed on each other, users can select their own individual layer(s) to work on while others simultaneously draw

on different layers. Drawing is permitted only on the topmost layer for any given user, though each person can have a different "top" layer. Items can be cut and pasted across layers. Layers can be treated as private scratchpads and later merged into the group's collective drawing stack. Users may work simultaneously on different segments of a single drawing layer, or they may work on different layers which, when combined, comprise a single complex drawing.

Testing showed that the transparency layers were often used as a means of identifying each individual's contribution; one layer was assigned per user. The authors point out that one drawback of using transparent layers is "layer overload". "At some point, too many designs with too many different patterns will overlap each other in the layered design space. It will be difficult for users to disambiguate the lines of one layer from that of the other." (Lu and Mantei, 1993, p. 96). Another drawback, discussed by the authors, is the difficulty in determining if two dimmed items belong to the same layer or to two different layers, particularly when there are many layers. The ordering of layers within the stack is also difficult to ascertain.

Other test results revealed that users were generally able to employ transparent layers as a means of managing their drawings. They often created different layers for each idea. They used layers to structure their work, keeping the base layer as the "common shared" drawing area. Layers were used for comparing different design options and for consolidating design pieces into a single drawing. (Lu and Mantei, 1993).

While the CaveDraw work described fairly extensive user testing results, its intent was not to evaluate permutations on the transparency factor itself. Results indicate both positive and negative attributes for transparent layers – using "dimming" as the mechanism for achieving transparency. Our experimental work provides detailed information about the levels of transparency which might be most useful and the types of information which can and cannot be easily combined on separate layers. As such, it should extend on the CaveDraw work and suggest further design alternatives for this system.

Transparency in Map Application

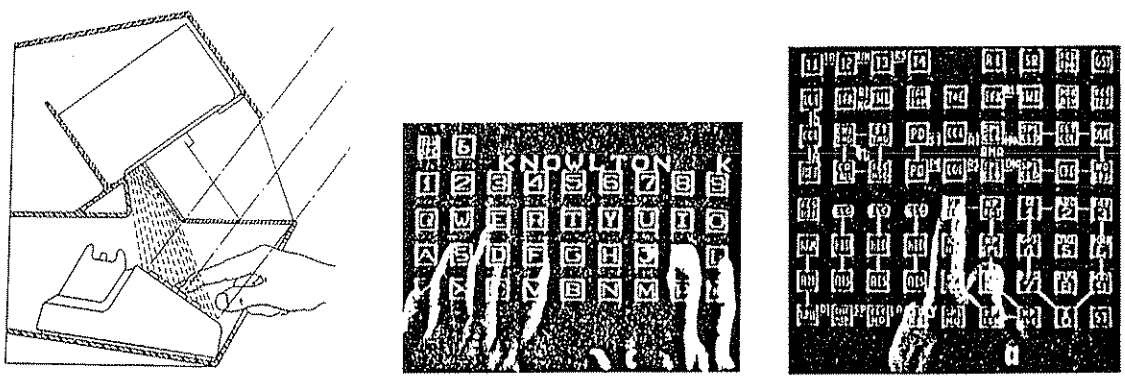
Another recent application of transparency was implemented by Lieberman (1994) for map reading applications. Demonstrations of this paper showed multiple layers of information on a map, where each layer was transparently superimposed on top of other layers. For example, on a regular view of a map there might be a region of 250 miles shown. Using a zoom operation, this system would transparently superimpose a 50 mile radial view on top of a selected city. The text labels, highway markers, and roads in this magnified view appear in the foreground, while the overview map appears in the background layer. While conceptually this sounds appealing, the

result is a highly complex combined image. It is obvious that visual interference is playing a significant role in the legibility of either layer (which at a glance appears almost impossible).

In Lieberman's brief paper, two other images are shown, illustrating other possible examples. A computer desktop is shown with transparent overlapping windows. One of the windows is magnified and appears superimposed on both the desktop and the other open windows. Again, the resulting image is highly complex – though it is clear that visual interference is slightly less of a problem when text files and graphical information are combined. Again, this research does not report any user testing or empirical results (Lieberman, 1996, personal communication). We believe that our systematic evaluation of transparency levels and of visual interference could be applied in the design of this system to enhance the concepts illustrated and improve usability.

Optically Superimposed Keyboards

This work by Knowlton (1977) used transparent layers albeit in a very different implementation. Half–silvered mirrors were used to project key cap labels down onto a blank keyboard(s) (Figure 18a). This allows a software program to dynamically change the keyboard layout depending upon the current application context (Figures 18b and 18c). The key cap labels appear not only superimposed on the keyboard but also on top of the users' fingers. While this might seem initially unsettling, Knowlton reports that it allows users to read the labels at all times since they are never obscured by fingers. Although this application is somewhat unusual, it does represent a novel application of transparency layered user interfaces. Several prototype systems were described in detail and were compared to more conventional systems. However, no user testing or experimental evaluation results were reported.



18 (a) schematic of system 18 (b) QWERTY keyboard overlay 18 (c) large keyboard overlay

FIGURE 18. Graphical overlay for keyboard (Knowlton, 1977, p. 369, 373)

Overlay of Input Devices and 3-D Display Space

This system developed by Schmandt (1983) follows a similar line to Knowlton's work. Half-silvered mirrors are used to project a display space image onto a surface, while the user can simultaneously see their hands and an input device. However, in this case, the input device is a wand (which was a predecessor to the Polhemus) and the display space is projected as a stereoscopic 3-D scene (Figure 19). The user moves the wand to select and interact with objects in the scene. The goal of this work was to design an interaction style that supported and maintained a more naturalistic spatial correspondence between a 3-D input device and the 3-D space it operated within. Through the use of mirrors, users could see both the input device and their hand within the work space. A detailed description of the system is given (Schmandt, 1983) but no evaluation was conducted (Schmandt, 1995, personal communication).

ImaginEasel

The last system we report on is one which uses half-silvered mirrors to present a user's hands within the context of an animated drawing (Figure 20). This system, developed by Disney, is currently being used in-house for animators. There are no publications available and no evaluation data are reported. However, the system does indicate the wide and diverse range of application domains that are interested in and pursuing various forms of transparent user interfaces to improve users' work.

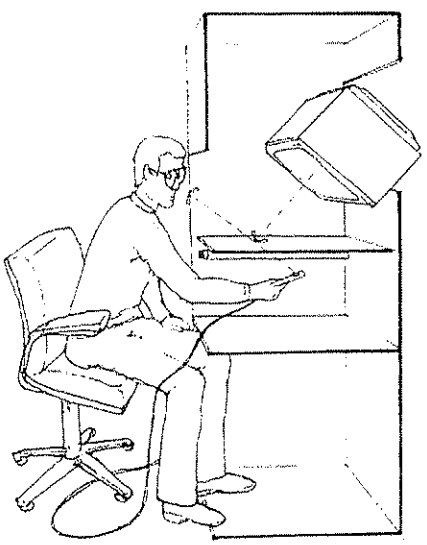


FIGURE 19. Input device for 3-D objects (Schmandt, 1983, p. 253)

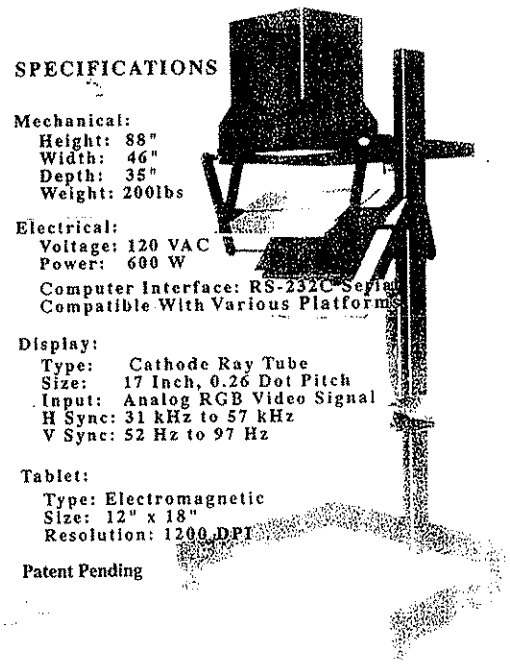


FIGURE 20. Disney's ImaginEasel (Walt Disney Corp., publicity brochure, 1994)

3.2 Heads Up Displays Systems

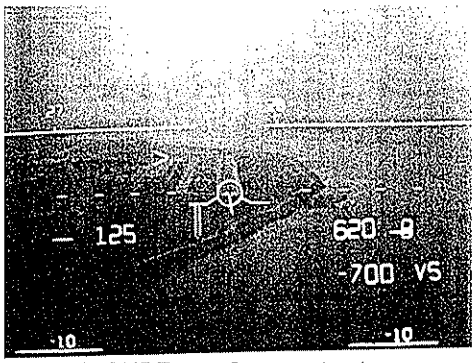


FIGURE 21. Contact Analog Heads Up Display Design (Desmond, 1987, p. 48)

The concept of a heads-up display (HUD) was developed in 1956 as a means of presenting airspeed and altitude information by superimposing aircraft instrumentation on the external real world scene (Naish, 1979) i.e., by reflecting images of the computer data on the aircraft's windshield (e.g., Figure 21). (Note that in more recent designs, HUDs also include computer displays projected on the inside of helmet visors.) There are two basic information presentation formats in HUDs. *Conformal formats* (developed by Lane, Cummings, Gold, and Pine in 1956) attempt to map real world objects and computer generated symbols in a one-to-one direct representation. The symbols overlay the real world objects (i.e., they share common contours) and they move in synchronicity with the objects. A sub–class of conformal format HUDs is a *contact analog* (developed by Klopfstein in 1966). These displays contain no digital information but rather present a "contact analog" of the environment and a format to directly assess various aircraft flight variables. For example, a contact analog might contain a symbolic representation of the horizon overlaid on the actual horizon or an overlay on the actual runway (see Figure 20). Some HUDs present partial conformal displays where only limited critical external world information is represented (e.g., a shortened runway or subsets of movement characteristics). In *non-conformal formats* the symbols are not necessarily the analog of the real world object. They may be abstractions or conceptualizations of an object or a function (e.g., airspeed) and the computer display may reflect a different scale than that of the external world. Conformal and contact analog displays offer the advantage of pictorial realism. They also integrate the instrumentation and visual information into a common representation. Non-conformal displays facilitate instrument and functional integration. Both conformal and non-conformal techniques are used in current heads-up display designs.

The key issue for the use and design of HUDs seems to be how well pilots can switch their frame of reference between qualitatively different stimuli represented by the computer-based information of the HUD and the environmental information of the external scene. This includes the ability to focus attention on one aspect of either display and also to divide attention between the two displays. "The goal of HUD design, then, is to facilitate parallel monitoring (divided attention) between the information in the near domain, defined by the HUD, and in the far

domain, defined by the environment; while continuing to allow the pilot volitional control of focused attention on one domain without interference from the other (selective attention)" (Larish and Wickens, 1991, p. 23).

The principle advantage of HUDs is that they minimize time spent looking at cockpit instruments within the aircraft and not looking at the real world scene. By superimposing critical information from the instruments onto the pilot's forward field of view, pilots can allegedly more quickly and easily monitor both information sources. Information can be displayed in several ways: it can be displayed directly on the windshield of the cockpit (differing perceptual depth from the external scene), or it can be *collimated* as an image presented on a combiner glass in front of the windshield, thereby appearing to float far outside the cockpit in the same depth plane as the external world (optical infinity). The argument in favor of collimated displays is that they allow the pilot to maintain the same optical focus for both information sources, thereby saving refocusing time. This seems to be the most popular design choice based on this argument.

The information displayed on the HUD is intrinsically different from the three dimensional real world view. The environmental information source contains texture, light, structure, motion and depth information. The HUD contains alphanumeric, graphical, and 2-D geometrical information (sometimes at blurred resolution with different types of 2-D depth cues). For this reason, it may be perceived as separate information regardless of the merging of optical distance of presentation for the two information sources (Fischer, 1979; Larish and Wickens, 1991). However, Naish (1964) found that by collimating these two information sources, they were treated as a single source of information. Collimation and superimposition theoretically minimize the necessity to physically shift focus and vertical fixation between two separable sources of information. Most research results seem to support the view that these are still treated as two separable information sources (similar to the finding of Neisser and Becklen, 1975; Becklen and Cervone, 1983).

There are several reasons why the literature on HUDs is of interest to us. First, for a restricted set of application domains where the two layers of information have similarities to HUD information, the experimental findings may be directly relevant. For example, one would anticipate that for graphical computer displays over video images, photographic stills, animations, or perhaps even drawings and solid models, the results of HUD evaluations will provide some insights into benefits and drawbacks. Secondly, this is one of the few areas of overlapping transparent display design where any empirical results are available. Using both the methods and the results, we can potentially infer significant performance aspects of focused and divided attention to use in our own experiments and designs.

Experimental Findings on HUD Performance

A number of HUD advantages have been cited including more accurate and efficient performance (Fischer et al., 1980; Greene, 1988; Naish and Miller, 1980; Newman, 1987a, 1987b; Steenblik, 1989), superior tracking (Naish, 1963), smaller touchdown footprints with smaller lateral and longitudinal dispersions (Stout and Naish, 1967), and improved landing precision (Desmond, 1987; Lauber et al., 1982). However, studies also raise a number of concerns and potential problems including distortions in optical effects on size and distance judgments resulting from collimation (Roscoe, 1987) and problems in attentional tunneling to the HUD symbology to the exclusion of the external world scene (Fischer et al., 1980; Hockey, 1986). The results on attentional tunneling seem most relevant to our particular application domain. These were discussed earlier when we described the consequences of focused attention in dual task environments.

Some of the most relevant and interesting findings relate to proximity comparisons between objects located in each display (the HUD and the real world scene). Studies investigating the effects of proximity (Cartesian distance) between displays elements on performance have fairly consistently shown performance improvements with increases in spatial separation. (McCann et al., 1993). Performance comparisons of Cartesian distance between elements suggest that items which are either too close (clutter) or too far apart (eye movement required) will degrade performance. If elements of one display can be integrated into the elements of the second display, forming a single perceptual object, these distance effects can be greatly reduced and performance will improve. It has been conjectured that spatial proximity seems to play a greater role than perceptual object belongingness and that more cognitive resources are needed to mental separate images when physical separation is absent (Larish and Wickens, 1991). There is some evidence that decreases in optical separation generate displays which are more "mingled in" thereby increasing decision time, presumably since separating out the necessary information is more difficult (Weintraub et al., 1984, 1985). These studies suggest that the time saved in visual accommodation is offset by increases in cognitive switch time to focus on one information channel.

3.3 Psychological Issues in Visual Attention

The field of visual attention is an evolutionary one, with many of the theoretical underpinnings still debated and contested as new experimental results challenge proposed models. Early investigators, primarily from the Gestalt and Behaviorist schools (e.g., Kohler, 1929), proposed that a simple set of rules such as isomorphism or conditioning could be applied to explain how

outputs (responses or precepts) related to inputs (stimulus or field). Although these two schools differed widely in their method of investigation, they both supported the notion that attention was the mechanisms which determined the significance of stimuli, thereby making it possible to predict behavior by these stimulus-response considerations alone. By the end of the 1950s, attention became a central research topic used to help explain the spontaneity and autonomy in behavior not addressed by behaviorism, Gestalt theory, or psychoanalysis. Behavioral unpredictability could not be explained by characteristics of the stimulus alone. In numerous cases, the organism seems to control its choice of stimuli out of the multiple possible sources, and hence, it determines how the stimuli might control its behavior. "The organism selectively attends to some stimuli, or aspects or stimulation, in preference to others" (Kahneman, 1973, p. 3). From this early attention research, theories were proposed which would account for this ability to seemingly select or ignore certain aspects of the stimuli. Our research interests are in these deliberate selective attention processes and, in particular, those related exclusively to vision.

The work by Treisman (1969) is a well-known and often cited taxonomy of selective attention operations. Attention tasks are classified according to what they require the subject to select: inputs or stimuli from a particular source; targets of a particular type; particular attributes of objects; outputs or responses in a particular category. According to Kahneman (1973), there was growing agreement that these varieties of selective attention are governed by different rules and are explained by different mechanisms. For selective attention, a number of factors play a role in determining why something is selected, how long it is selected for (i.e., attended to), and how many things can be selected (i.e., how attention is partitioned across multiple items).

Aspects of attention include not only selection itself but also amount and intensity. At certain times in the day, we seem to experience an overall drop in our level of attentiveness. This is not merely a failure to pay attention to one thing in favor of another – it is that we have less attention to give to anything. Intensity is related to levels of arousal (including novelty, complexity, and incongruity) (Berlyne, 1951, 1960, 1970). Intensity is also related to effort, which is much more than merely wakefulness; effort invested corresponds to the task demands in what the subjects are doing (i.e., Yerkes-Dodson Law). Finally, intensity and arousal vary according to whether attention is voluntary or involuntary. Voluntary attention is an exertion of effort in response to the selection of current plans and intentions i.e., some stimuli are chosen because they are more relevant to the task at hand. Involuntary attention is an exertion of effort in unplanned response to more intrinsic stimuli, such as sudden noise or novel and surprising stimuli which spontaneously attract attention (Kahneman, 1973). We are interested primarily in voluntary

attention. However, there are occasions when the stimuli forces involuntary attention (such as in the Stroop Effect explain later).

Within the realm of voluntary attention, multiple stimuli can simultaneous compete. Research indicates that the human capacity to attend to multiple sources of information is limited. Most attention theorists assume a bottleneck somewhere in the system, but the cause of the bottleneck has been controversial. It is still debated whether attention is a unitary resource or a divisible resource. Certain activities can clearly be attended to at once (e.g., walking along a path and speaking), while at other times only a single activity can occur at one time. Each of these bottleneck theories has implications for the model of attention proposed. Several influential theories are summarized below.

Filter Theory (Broadbent, 1957, 1958)

The filter theory assumes that only one stimulus at a time can be perceived. When two or more stimuli are presented simultaneously, one is perceived immediately, while the sensory information about the other(s) is temporarily held as an unanalyzed image or echo (Figure 22a). These unanalyzed sources can only be perceived after perception of the first message is completed.

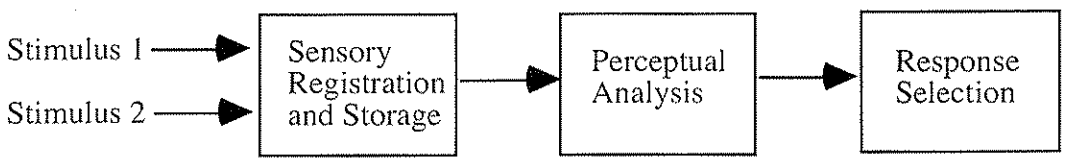


FIGURE 22 (a). Schematic of Filter Theory Model

Response Selection Model (Deutsche and Deutsche, 1963)

In the response selection model, multiple inputs may be simultaneously perceived; the bottleneck is located at the response selection stage. The meanings of all stimuli are extracted in parallel without interference, but the later bottleneck prevents the initiation of more than one response at a time (Figure 22b). The response that best fits the situational requirements is selected and processed.

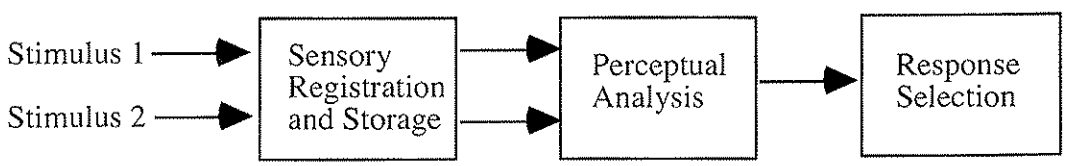


FIGURE 22 (b). Response Selection Theory Model

Capacity Model (Moray, 1967)

The capacity theory assumes that the bottlenecks in attention are attributable to a person's capacity to perform mental work. This limited mental capacity may be freely and flexibly partitioned amongst the concurrent activities, but it is a finite resource. This theory must deal with three central questions:

(1) what factors make an activity more or less demanding
(2) what factors control the total amount of capacity available at any one time
(3) what are the rules of the allocation policy

Changes in any one activity's demands result in a re–allocation or adjustment in attention or, in cases of overload when total capacity is exceeded, performance will deteriorate. Moray's model assumes that interference exists when such an overload occurs. It further assumes that interference is non-specific (as opposed to the earlier models which assume interference is based on the extent to which multiple activities require the same mechanisms). With a capacity model, concurrent activities which require attention tend to interfere with one another. The allocation policy under such circumstances is controlled by four factors:

(1) dispositions which reflect the rules of involuntary attention (e.g., sudden noise, movement, novel stimuli, your name in a conversation)

(2) momentary intentions (e.g., listen for voice to your right)

(3) the evaluation of demands; a rule that when activities require more than the available capacity, one activity is completed

(4) effects of arousal, especially in the high arousal case.

All theories reflect that there are attentional limitations and interference results when we attempt to process concurrent activities or stimuli. The first two models assume that this interference is a result of competition for similar perceptual or modal resources. The capacity models assume interference depends upon the demands of each of the tasks.

In our experiments on transparency, we exclusively examine visual stimuli (no multi-modal inputs), and either verbal responses (Chapter 4 - Stroop Experiments) or motor responses (Chapter 5 - Menu and Icon Selection Experiments), based on attending to a given target. Two competing images may create interference, where the attributes of the individual images make the task more or less demanding. We do not control or manipulate levels of arousal. We believe

that our dual visual tasks reflect an attentional trade–off best represented by a capacity model, such as that proposed by Moray (1967). This model does not specify the elements involved in the sequence of attentional processing, but rather it proposes a framework for understanding changing task demands, allocation of attention, and the trade-off based on a finite amount of attentional resource. Additionally, visual attention is necessarily comprised of both the basic psychophysical perception of an object and the cognitive processing which imparts meaning or semantics to that perceived object. Based on these semantics, a response is determined. Each of these components, perception, processing, and response selection, are part of the overall visual attention paradigm we use throughout this dissertation. It is this higher level, overall performance construct that we are interested in and that we subsequently measure.

3.3.1 Selective and Focused Attention

When there are multiple sources of possible information to choose from, we must periodically select the information source that we believe to be the most important or relevant to the task at hand. Occasionally we make errors in choice, attending to an inappropriate source of information. These are conscious choices but may be inadvertently influenced by the characteristics of the information source. For example, brighter colors or motion may draw our attention to one object when we should be monitoring another. Limits in selective attention result from *an intentional but unwise choice of input* (Wickens, 1991). Focused attention is the ability to channel attention to one source of information in the environment. Problems in focused attention are the tendency we have to be distracted or unable to concentrate on this single source of information. The difference between failures in focused attention versus failures in selective attention is that in selected attention cases there is an intentional but unwise choice to process non-optimal information. In focused attention failures, *the processing of non-optimal information is forced by environmental events* (e.g., extraneous input - light or noise) despite the subject's best efforts to shut them out (Wickens, 1991). In fact, while Wickens attempts to clearly delineate a distinction between focused attention and selective attention based on intent, few researchers seem this careful about terminology. At best, selective attention implies voluntary action and focused attention seems to be the deliberate concentration of this voluntary action on a single item of interest. We rely on the more precise definition, which suggests that focused attention is central to our research in interference and visual attention. The task goal or user intent in our case is clear at the outset. However, interference can arise as part of the involuntary processing of conflicting information which cannot be ignored.

Selective attention models assume that the environment is divided into channels of information and that there is some "optimal" way of monitoring these multiple channels. Optimality implies

that performance reduces potential costs and maximizes potential benefits. In general, selective attention can be almost perfectly effective when guided by an appropriate cue (Kahneman, 1973). In cases where a particular target must be found, one would anticipate that pre-determining the target and knowing its attributes in advance would result in high levels of performance. However, selection is effective only when the relevant and irrelevant inputs differ in obvious physical characteristics (Ibid.). Focusing on one object does not prevent the processing of information from irrelevant objects, but the extent of this interference relies heavily on the similarities and dissimilarities between the two. This is clearly consistent with the Gestalt models of object and figure recognition and discrimination and relates directly to our problem of overlapping transparent layers and the potential for interference between layers.

3.3.2 Divided Attention

In divided attention, we are interested in attending to multiple objects simultaneously. Failures in divided attention therefore are failures to divide our attentional efforts between the multiple sources of information (time-sharing), all of which we need to process. If the task requirements, stress levels, or workload are too demanding, we fail to divide our attentional resources adequately to process the multiple inputs. Divided and focused attention can be defined in terms of a channel model (Treisman, 1969). Our attention can be focused on a channel of information where all events within this channel can be roughly processed in parallel. Simultaneous events within a channel must be processed in parallel; we cannot block one or the other out. Information which appears in separate channels must be processed serially, forcing us to block one channel while focusing on another.

A number of characteristics can be used to define a channel. In essence, the more similar two "signals" or information sources seem to be, the more likely that they will be perceived as a single channel. A channel could be argued to be a perceptual object which supports a number of the attributes and properties proposed originally by the Gestalt psychologist. For example, the pitch or tones of sounds, the gender of voices, the size and font type of text, or the proximity of graphics – all define whether two items are perceived as part of a single information source. The most obvious property that defines a channel is space - proximal locations in the visual field. It is difficult to focus on one source of information coming from one location in space when a second kind of information is delivered at that same location.

The research described in this dissertation looks at a person's ability to focus attention on either the foreground layer or on the background and the performance trade-off between the two. Supporting some level of dual task awareness is inferred to be pre-requisite to support divided

attention. This might be achieved through processing information in parallel or through rapidly switching between the two layers. In either case, we are interested in the switching cost and the ability to correctly discriminate between items belonging to one layer. We believe that a defined range of transparency can be found in which people can effectively focus attention on either a foreground task or a background task and can switch quickly between these tasks. This dissertation examines the focused attention on foreground and background in detail through controlled experimentation. We examine issues related to switches of attention as they relate to a real product scenario (Chapter 7 - Case Study). However, an in-depth investigation of divided attention problems and phenomenon are outside of the scope of our current research program.

3.3.3 Space Based and Object Based Theories of Attention

Two main theories have been proposed to account for the manner in which attention is distributed in the visual field: space based and object based. Space based theories (e.g., Eriksen and Eriksen, 1974; Eriksen and Yeh, 1985) suggest that analogies like the spotlight metaphor, zoom lenses, and gradients best represent the way in which visual attention is distributed in contiguous regions (not unlike bulls-eye targets). Stimuli that fall within the "spotlight" are processed in detail while items outside this region are ignored. This theory suggests that conflicting or interfering stimuli to some designated target stimuli will produce dramatic performance penalties when the two are within about 1 degree of visual angle (e.g., Eriksen and Eriksen, 1974). Clearly, for divided attention, optimal performance would occur when the two stimuli are aligned spatially within this visual region (e.g., Kramer, Wickens, and Donchin, 1985). The spotlight metaphor was modified to account for some degree of graduated awareness over wider visual regions, giving the zoom lens analogy (Eriksen and Yeh, 1985). (Zoom lenses provide high levels of detail at a central point, with gradually fading detail and size according to the distance from this center.)

Object–based theories do not rely on proximity but instead are based on the more Gestalt theories of perceptual organization using contour, color, movement, etc. as defining characteristics of attentional allocation. This model suggests that focused attention necessarily processes *all* attributes of a single object; different attributes of a single object are processed in parallel, while different objects are processed serially. For focused attention tasks, this means that the extent to which conflicting or interfering information can be located on different perceptual objects determines performance improvements. For divided attention tasks, integration of information on a single perceptual object whenever possible will facilitate its perception and hence will improve performance. This allows multiple stimuli to be tightly integrated into a single precept

as opposed to remaining in independent sources, hence perception of this single integrated object is more rapid (compared to acquiring information from the independent sources).

Kramer and Jacobson (1991) point out however, that these two theories of attention are complicated by the fact that "two objects are usually located at greater distances from each other than are two properties of a single object" (p. 268). In effect, objects and space co-vary. They attempted to control for this in their studies of space–based models versus object–based models. Their results are consistent with the object–based theory – response time is least with targets and distracters on the same object than when located on different objects (controlling for proximity). However, they also found support for the space–based model where proximity may be one of the key attributes that determines a perceptual object. Finally, the study concluded that designations of objects, per se, seems insufficient to predict focused attention effects. It remains unclear how to categorize the strength of attribute groupings used to determine within–object and between–object similarities. Clearly, what constitutes these attributes and their respective unity is central to object perception.

Attempting to apply these theories to our research is complicated by the covariance between space and object. In our case, we are necessarily concerned with a scenario created by the superimposition of one transparent object or layer over some underlying information. It is clear that the combined information sources are proximate to within 1 degree of visual angle. Therefore, we are investigating the case where space–based models would have maximal interference between the two layers (worst case scenario). The extent to which object–based theories apply will depend upon the attributes that are considered to form a single object and whether these attributes appear in our task domain or not. We have only limited control over manipulating the information content of the layers to maximize the dissimilarity between foreground and background. Based on an understanding of object attributes, our subsequent experiments do manipulate the types of information in each of the combined layers.

3.3.4 Object Perception and Recognition

The Gestalt approach to attention and perception (Kohler, 1929) suggests that a grouping occurs very early in processing stimuli (see Figure 23). This implicitly segments or groups sets of stimuli together that form the same unitary distinct object in the scene (*unit formation*). These units have both temporal and spatial aspects: spatial aspects define perceived objects while temporal aspects define perceived events. Following this grouping, attention comes into play, determining which items receive *figural emphasis* (attention is allocated to objects). These attended objects are more likely to be perceived consciously and in more detail than other

objects. *Recognition units* are then activated based on matching features of the object with an image stored in memory. The accuracy of this depends upon the intensity, resolution, etc. of the object perceived. A perceptual interpretation is then applied to the object based on the matching features which correspond to this internally stored interpretation (*selection of interpretations*). Only one such interpretation is assigned to any given object. Based on this, a response is initiated (*response selection*). This process of perceptual analysis is presented in schematic form by Kahneman (1973, p. 67), and shown below. Attention is allocated at two stages: figural emphasis time and response selection time. Attention is most easily directed towards natural perceptual units or groups. Some characteristics which facilitate visual groupings are similarity in shading, shape, proximity, size, patterns, slope, motion patterns, color, or brightness.

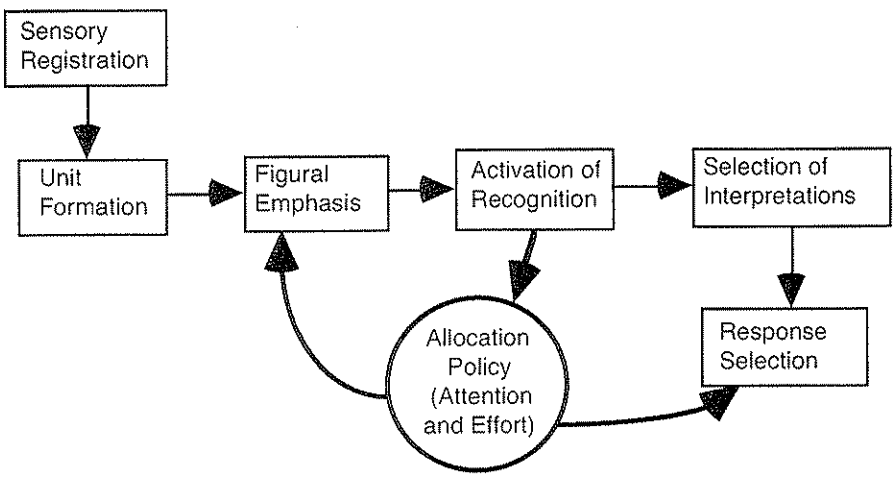


FIGURE 23. Schematic Model of Perception (Kahneman, 1973, p. 67)

The above breakdown allows us to separately consider the processes which contribute towards stimulus selection (i.e., selecting which stimuli are relevant and irrelevant based on physical characteristics) and those which contribute towards response selection or determining which response is required given some specified set of possible alternatives. Our research focuses on the processes related to stimulus selection (as opposed to response selection). We assign targets which must be chosen from a set of candidates, given variations in interference (visual competition). While we are not conducting analyses which precisely determine the components of item recognition, clearly the ability to infer object boundaries (where one object begins and ends) and to differentiate objects has an impact on performance. For this reason, we briefly highlight research on the most relevant aspects of object recognition below.

In our research, we wish to maintain object consistency within one layer while minimizing the extent to which interference results from objects in a second transparent layer. The work on object recognition suggests that dissimilar attributes are less likely to be integrated into a single

unitary object, facilitating layering, despite transparency. For example, two objects of different colors or textures are more likely to be perceived as two distinct objects and not one integrated object, even when overlaid. We anticipate less obvious object boundaries and hence more interference in cases where there are similar attributes in each of the overlapped transparent layers (e.g., items of the same color, same contrast level, same texture, same size, etc.). This is discussed in more detail in subsequent chapters in which we present our specific experimental results.

In addition to attributes which are used to determine object boundaries, a similar process seems to apply when deciphering "figure versus ground" (*figural emphasis* in Figure 23). The figure is generally taken to mean the object which appears to subjectively stand–out and predominate in the scene. Figures have bounding contours which belong to the figure rather than to the background and there is a depth effect where the figure appears closer to the observer than the "ground". Elements which are perceived as belonging to the figure are more easily perceived and remembered than those which apparently form part of the ground (Weitzman, 1963; Kahneman, 1973). Some of the factors which tend to be used to distinguish figure from ground are: smaller size of an object, warmer colors (red, yellow), moving objects, more contour rich scenes or objects, isolated objects, higher brightness, dissimilarity between objects, and specific target identification intent. The decision to select some stimulus for special inclusion can be made before it is visible. This has immediate effects on how it is perceived (i.e., there is a bias toward that stimulus being perceived making the figure it is a part of stand out) (e.g., Neisser, 1967). This is particularly relevant for our studies, where stimuli are known in advance. The time it takes to search for a pre-determined target is largely influenced by its discriminability. This is discussed in the context of our specific experiments and targets in the relevant subsequent chapters.

More recent work based on object perception has been conducted in the area of instrument display designs. Wickens and André (1990) found that spatial proximity had little effect on focused attention, whereas a distinct color code improved focused attention. However, upon further investigation, they concluded that spatial proximity to irrelevant items disrupts the operators' ability to locate relevant items. Color coding compensates for this. They used a variable, "objectness", which they formed by combining two dimensions into a single object. This objectness disrupted focused attention but improved information integration. Again, the use of color restored the focused attention accuracy. This suggests that some object–based attributes seem to provide stronger cues than others for object integration and that these stronger attributes can compensate or override the negative consequences of conflicting weaker attributes. Color appears to be one such strong attribute.

3.3.5 Attentional Tunneling

In addition to involuntarily processing attributes because they form part of a shared perceptual object, we also consider the attentional tunneling phenomenon that sometimes results from focused attention tasks. Attentional tunneling has also been called "cognitive capture" (Weintraub and Ensing, 1992). Attentional tunneling results when a person focuses attention on a single source of information and all other information passes unnoticed – even that which is directly superimposed or in visual alignment with the focus of attention. This tunneling is particularly problematic for tasks which require divided attention. It has been extensively studied in the field of Heads Up Display (HUD) design evaluations (e.g., Naish, 1979; Naish and Miller, 1980; Larish and Wickens, 1991). Many of the hypothesized causes are directly related to how information displays distract operators or how operators group information (in effect creating perceptual object boundaries).

Differences in information format may affect the tendency to fixate on or be compelled by the symbology of the display over the real world view (Naish, 1964; Naish, 1979; Naish and Miller, 1980). Conformal and contact analog displays may use symbology that does not require the pilot to use real world visual information to assess the flight path. This increases the tendency to fixate on the HUD information, ignoring the external world (Naish, 1979; Fischer et al, 1980). This seems, in part, to depend upon the specific design of the symbology (Greene, 1988; Lauber et al., 1982). High levels of stress or workload seem to enhance this attentional tunneling. Findings suggest that an excess of symbology on the HUD may obscure the external scene and contribute to higher workload (Fischer et al., 1980; Gold, 1968; Newman, 1987a, 1987b; Weintraub et al., 1985). Larish and Wickens (1991) propose that the mechanisms creating conditions of involuntary fixation to HUDs are more than an issue of clutter or glare. One explanation is the rapidly changing, dynamic nature of HUD information which requires more continuous monitoring by the pilot.

This research has several implications beyond those discussed earlier regarding perceptual object groupings. In our applications of transparent layer user interface tools, we must be aware of distractor elements (such as flashing words or moving icons) and of the amount of information displayed. We can anticipate that large quantities of information displayed on a partially transparent foreground layer would have the potential to create attentional tunneling. Rapidly changing information appears to move or flash, thereby directing attention to itself on an on-going basis. This suggests that the nature of the tools and tasks need to be considered to appropriately apply transparency.

3.4 Selective Looking in Dual Task Experiments

Research in selective looking (e.g., Kohler, 1972; Neisser and Becklen, 1975; Becklen and Cervone, 1983) suggests that using transparency is a promising method of presenting foreground and background information. However, a major concern is the extent of visual interference between the two layers or information sources. Below we present the relevant studies using transparent overlaid displays.

Kohler (1972) originally investigated selective looking (monitoring dual tasks) by building headgear using half-silvered mirrors which presented the scene of the world in front of him superimposed on the scene of the world behind him. He reported that he could easily switch between these two views; the unattended scene seemed to "disappear" from sight. In this report, there was no specific measure of the possible optical distances between views. Therefore, it is unclear to what extent the scene differentiation can be attributed to accommodation of the lens of the eye versus other perceptual or cognitive processes.

Motivated by this work, further studies were carried out (Neisser and Becklen, 1975; Becklen and Cervone, 1983) using two superimposed video images presented on a single monitor. These studies investigated how difficult or easy it was to follow one scene and ignore another when both are presented at the same optical distance (thereby eliminating lens accommodation). They also investigated how well subjects could monitor both scenes simultaneously, if required to do so (again with the scenes presented as combined video at the same optical distance).

In the first study (Neisser and Becklen, 1975) the tasks were visually distinctive: a hand slapping game and a ball tossing game. In the later study (Becklen and Cervone, 1983), both tasks were visually similar ball tossing games; the tasks were differentiated by the color of the shirts worn by the players. In both cases, subjects were asked to monitor one task and indicate the irregular occurrence of target events in this task. Meanwhile, bizarre events were sporadically presented in the non-monitored task. Subjects were easily able to monitor the target task to the exclusion of the unattended task. Subjects did not notice the bizarre events, even when the experiment was stopped during, or immediately after, the bizarre event occurred and the subjects were asked about it. This result still held when the bizarre event was presented in the exact same visual location where the target event occurred (i.e., within foveal range). This seems to indicate that the intentionally unobserved task went virtually unnoticed. A number of alternative explanations for this phenomenon were discussed and discounted. Neisser and Becklen (1975) speculate that the primary factors that distinguish one scene from the other were the intrinsic properties and structure of the scenes and not the distance or clarity. (In their studies both images were

superimposed with no variation in transparency – it was set to 100% clear, in the context of our criteria). "What is seen guides further seeing" (Ibid., p. 491). Through follow–up studies, Neisser and Becklen discounted eye movements or foveal status as the primary mechanisms in selective looking behavior. They further claim that the unattended scene does not really "disappear"; subjects seem to remain always aware that something else is occurring although what specifically is occuring is not really seen. The results of these studies suggest that two superimposed video tasks can be easily separated out and one can be monitored with minimal interference from the other.

Another condition had subjects simultaneously monitor two scenes and indicate the occurrence of target events in each scene. A "very drastic deterioration of performance [was observed] when subjects were asked to monitor both episodes simultaneously" (Neisser and Becklen, 1975, p. 491). Subjects missed 20% to 40% of the target events and numerous false alarms occurred. The extent of simultaneous task awareness is unclear. The ability to monitor simultaneous superimposed video sources seemed to be limited more by the skill and practice of the observer – "the more skilled the perceiver, the more he can perceive" (Neisser, 1976, p.93). This suggests that even the more difficult task of divided attention or time sharing can be achieved, though practice may be required.

This previous research, though not applying directly to graphical user interface design, suggests that the use of superimposed transparent displays is promising. Based on these results, one would anticipate reduced switching times and potentially improved awareness by minimizing head and eye movement and re-focusing (or re-accommodation). This would be particularly true in cases where users acquired high levels of skill or expertise. Also, one can reasonably anticipate that users will be able to treat the visual scenes separately and voluntarily attend to one or the other (with varying degrees of interference). These studies also suggest possible drawbacks . As in most interface designs, one can anticipate some inappropriate applications and pitfalls as well. In cases where missed observations in the unattended layer have a high cost, reducing visibility through transparency might be undesirable. Also, if both tasks must be simultaneously monitored and both have high attentional demands, attentional tunneling problems (such as those encountered in HUD design) might arise. Finally, we need to further understand the role of visual interference as it relates to transparency levels and to information content (i.e., visually similar and dissimilar information types).

3.5 Visual Interference and the Stroop Effect

It is almost impossible to discuss visual attention and interference without noting references to the Stroop Effect. The original Stroop experiments (Stroop, 1935) provided intriguing results about the nature of focused attention and interference in a dual task scenario. It is virtually impossible to consciously block or prevent the Stroop Effect in selective looking tasks, despite numerous experimental permutations (over 700 articles to date – for reviews see Jensen and Rohwer, 1966; MacLeod, 1991).

In traditional Stroop tasks, a series of words are presented in randomly chosen ink colors (e.g., red, green, blue, yellow). There are two task components: color naming and word naming. Subjects must name the ink color while ignoring the word. Some of the words are neutral (e.g., uncle, shoe, cute, nail); other words are the names of conflicting colors (e.g., yellow, blue, green, red). Consistent, significant performance degradation occurs when conflicting color words are used and subjects attempt to name the color of the ink (e.g., the word "red" appears in green ink; the correct response is green). Subjects are unable to consciously block this interference out. However, the word naming task is relatively insensitive to any interference from the colored ink. In the word naming task, color conflicting words can generally be read just as fast as neutral words. These results have generated enormous discussion about the nature of visual attention, pre-attentive processing, and the semantic perception of words.

The Stroop experiments provide an extremely sensitive measure of interference for dual attention tasks and, as such, we wish to apply these tasks in our research by introducing a transparency condition. This should provide us with conservative and stringent boundaries limits for focused attention and legibility, given variations in transparency level. Several key results provide the groundwork to justify our particular use of the Stroop Effect. First, we wish to split the Stroop dual attention task into perceptual layers so that we might manipulate the transparency of the topmost layer. We have therefore modified the original Stroop experiment to be a color patch *through which* a black text word is seen (as opposed to using colored ink to write the word). Justification for this alteration is partially supported by two prior publications which most directly relate to this particular permutation of the original Stroop tasks. Kahneman and Chajczyk (1983) found a consistent and significant Stroop effect when the word was printed in black ink, presented adjacent to a color bar. In their experiments, colored bars were presented above or below a black text word without dilution to the original Stroop Effect. They used angles of eccentricity between the word and the color patch as a further independent variable, testing 2 degrees and 4 degrees of eccentricity. The conclusion was that the Stroop Effect "occurs even when the interacting stimuli are very far apart" (Ibid., p. 507). Another highly relevant Stroop

permutation was done by Kamlet and Egeth (1969), who conducted the color naming experiment using white text presented on colored tape. The white text contained the names of colors (e.g., red, green, etc.) Subjects were to name the color of the tape. Again, results that were consistent with the traditional Stroop Effect were obtained (i.e., color conflicting words produced reduced verbal response times). Given the above results, we anticipate that using a color patch with a black text word should still produce results consistent with the original Stroop experiments for the color naming task.

The second component of the Stroop experiments, the word naming task, is clearly affected by legibility. Manipulating a transparency variable can be expected to produce a direct performance degradation on legibility (i.e., when reading through the color patch, as the patch becomes more transparent, legibility and hence reading becomes easier and faster). There are few relevant studies on variations for legibility and word naming as it relates to hue, saturation, or brightness. One study investigated the effect of hue variation (Dyer, 1971) for the text letters but not in terms of word naming and legibility. For color naming tasks, chromatic patches showed more interference than achromatic patches, when color conflicting names were used. Work by Gumenik and Glass (1970) and Dyer and Severance (1972) further measured word reading time for colored and achromatic words. (Note that neither of these studies conducted the color naming task component.) Results of these studies suggest that the presence of colored ink instead of achromatic (gray) or black, high–contrast ink influenced the speed of word reading (i.e., slower response times). Sichel and Chandler (1969) were interested in potential clustering effects where subjects might inadvertently group stimuli based on "(a) the particular word combination, (b) the color and brightness of the stimuli, (c) the interaction of color and word meaning, and (d) variations in perceptual span" (Ibid., p. 220). Their results related to color and brightness showed that for either colored rectangles or colored ink, verbal response latencies differed as a function of both color and brightness of the stimuli. In particular, red showed faster response times than blue or green. However, they further suggest that the magnitude of the effect was moderated by the hue and brightness, and that the latencies were not the same for all word by color combinations. One difficulty in comparing existing literature is the omission of any data which specifies the precise colors that were used. No measures of frequency, wavelength, or more recent measures such as RGB and HSV are given. As MacLeod (1991) states, this is clearly an unexplored area, and therefore it is "premature to offer empirical generalizations".

The above studies suggest that we can expect the separation of word text and colored patch to have minimal effect on the overall Stroop interference results for color naming. The impact on word naming as it relates to hue, brightness, or saturation level is unknown. In general, and in accordance with the original Stroop word naming results, we anticipate word reading to be

relatively stable and insensitive to interference. However, we deliberately add a twist to the traditional Stroop paradigm by investigating how degree of transparency affects the interference between the displayed word and the color target. This obviously has a direct impact on word legibility for word naming performance and, as a consequence of the reduced legibility, should reduce interference in the color naming task. Finally, despite the fact that over 700 articles have been published on the Stroop effect (for reviews see Jensen and Rohwer, 1966; MacLeod, 1991), the impact of this particular manipulation has never been investigated. Therefore, in addition to contributing towards the solution of an applied problem, the Stroop experiments described in this dissertation also make a unique contribution to the basic research literature on the Stroop Effect. In particular, this represents a unique application of the Stroop effect for applied research to determine aspects of legibility and interference as they relate to transparency.

3.6 Summary

In this chapter, we reviewed the relevant literature in focused attention, divided attention, and basic attentional theories to help us structure our subsequent experimental research. We discussed how our research fits into and compliments the existing body of work in the design of current transparent systems or interfaces. As such there is little evaluative work done thus far, providing us with an opportunity to contribute to the basic literature while helping to provide design guidelines. We highlighted findings from studies on selective looking with dual, superimposed transparent displays which suggest that subjects can separate out the two tasks. As a means of measuring visual interference under stringent conditions we described a permutation to the standard Stroop Effect using transparency as a new independent variable. Our Stroop-based experiments are described in the chapter that follows.

4.0 The Stroop Effect Experiments[1]

Our initial goal was to obtain an estimate of the semi-transparent level which represents the best compromise solution to the conflicting goals of supporting focused attention on the *foreground* layer versus supporting focused attention on the *background* layer (thereby supporting divided attention). A reasonable approach is to conduct this research under very conservative, stringent conditions. In this way, we can be sure that the estimate we derive is a lower bound for the desired solution. Accordingly, we chose to conduct our first three experiments with the strongest, most consistent method of investigating task interference known in the psychological literature – the Stroop (1935) Effect, described in detail in Chapter 3.

In the original Stroop Effect (Stroop, 1935), the word appeared *typed* in colored ink. For our experiment, we use a color patch and a black colored word which appears *through* the patch (Figure 24) while the color patch varies in opacity. We ran both the color naming experiment (interference sensitive measures) and the word naming experiment (legibility sensitive measures). When combined, results from the two experiments should help identify interface design parameters where interference is minimized and the word is still fairly legible (i.e., awareness is preserved). This conservative estimate of the optimal trade-off point can then be further evaluated in the context of more representative interfaces in realistic applications.

4.1 Experiment I - Color Naming

In this experiment, we were interested in verbal response times and error rates for a color naming task, where a colored rectangular patch and a word were simultaneously presented to the subject. In this case, the word is seen by looking "through" the color patch. This is roughly analogous to looking through a piece of colored acetate at a word underneath. The transparency/opacity of the color patch was varied such that the word was sometimes clearly visible, sometimes faintly visible, and sometimes not visible at all (only the color was shown). By being required to name

1 Portions of this chapter were published as Harrison, B. L., Ishii, H., Vicente, K. J., Buxton, B. (1995). Transparent Layered Interfaces: An Evaluation of a Display Design Space to Enhance Focused and Divided Attention. In the *Proceedings of CHI'95*, May 6-12, 1995. Denver, CO. New York: ACM Press. 317-324.

the color of the patch on the top layer, participants were engaged in a foreground focused attention task, albeit a very simplistic one. A standard Stroop paradigm was used to measure if the level of transparency dilutes the degree of task interference caused by the word (see sample stimuli in Figure 25). Based on the framework proposed in Chapter 2 – Figure 9, we anticipated that subjects will experience high levels of interference from the background word at high levels of transparency (e.g., 100% - clear), when they try to name the color (difficulty in focused attention on the foreground color). As the color patch becomes more opaque, the interference from the background word should decrease (making focused attention easier). The results of this experiment should thereby reveal the parameters of the focused attention foreground curve in Figure 1. This color naming experiment represents an extremely sensitive foreground focused attention task. As such, if the word is at all legible, some performance degradation is expected to occur. As a consequence, legibility is directly related to interference and hence to the anticipated performance results.

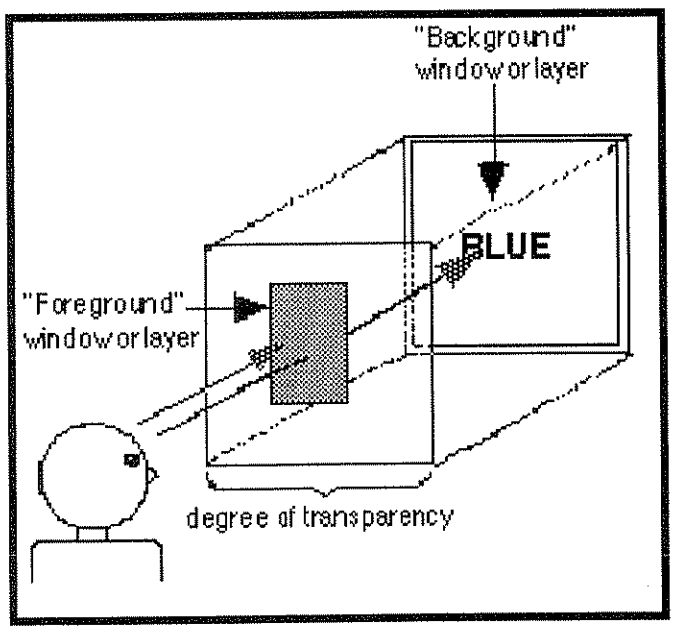


FIGURE 24. Model of Stroop Experiment Configuration

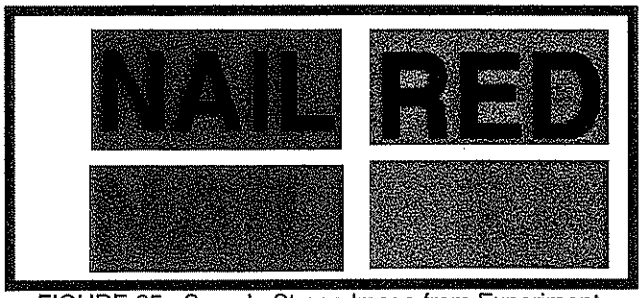


FIGURE 25. Sample Stroop Image from Experiment, letters appear in shades of black/gray depending upon transparency level, color patch appears surrounding letters

One item to note about the Color Naming experiment is the nature of the interference. The interference results from a conflicting color name with the color perceived. This constitutes *semantic interference* (i.e., interference is based on the meaning of the word). However, in our transparency permutation of the Stroop Experiment, the word's *legibility* determines whether this semantic interference can occur. If the word is not discernible then clearly no interference, visual or semantic, can occur. The letters must be readable before the word meaning may be inferred, resulting in the potential for semantic interference. Through our application of a transparency variable, we have directly related legibility to this semantic interference. This relationship allows for a more robust comparison between this experiment and the legibility component of the Word Naming experiment described later.

4.1.1 Hypotheses

H1: As transparency level increases (i.e., the word is more visible through the color patch) the response time and errors will increase (implying more interference), for the color naming task (as shown in Figure 9).

H2: Overall, there will be a significant difference in performance (errors and response time) for word types, within transparency level, reflecting the predicted Stroop Effect. The order from worst performance to best will be: incongruent words, congruent words, neutral words, and the baseline color only condition (blank - no word).

H3: Transparency will dilute the Stroop Effect. At low levels of transparency there will be minimal interference (word is virtually illegible) and hence no Stroop Effect. At higher levels of transparency the Stroop Effect will become stronger (i.e., the difference between word type groupings is more pronounced).

4.1.2 Method

4.1.2.1 Subjects

A total of 16 students from the University of Toronto served as subjects. They were pre-screened for color deficient vision. In pre-tests, all subjects could easily discriminate the different color patches used. All subjects had 20/20 or corrected 20/20 vision. Subjects were paid for their participation and could voluntarily withdraw without penalty at any time.

4.1.2.2 Experimental System Configuration

The software for this experiment was designed using the PsyScope development environment and Button Box hardware (Cohen et al., 1993), combined with a headset microphone on a Macintosh IIfx[2].

We purposely wanted to use verbal responses if possible, since motor responses, such as mouse clicks, have been found to dilute the Stroop Effect (White, 1969; MacLeod, 1991). The PsyScope hardware and a headset microphone addressed these problems. Microphone input levels were precisely adjusted for each participant by adjusting noise filters and volume pick-ups. Verbal response times from sound input onset to data recording were within +/- 1 msec of accuracy using the PsyScope timing hardware.

Subjects sat at a fixed distance of 100 cm from the screen. A 13-inch color monitor was used to present all stimuli. The Macintosh was equipped with a 24-bit color video board to allow precision color display of the stimuli.

4.1.2.3 Experimental Design

The independent variables were Transparency (5 levels), Word (3 levels), and Color (4 levels). The dependent variables were verbal response time (measured in msec) and error in response.

A 5 x 3 x 4 (Transparency x Word x Color) fully randomized, within subject, repeated measures design was used. All subjects were exposed to all possible combinations of Transparency x

---

2 Without this external hardware, we found high system response time variability occurs, based on the current activities that the operating system is carrying out, such as memory management. At best, if the operating system is idle, 17 msec accuracy can be achieved on a Macintosh. We additionally programmed and tested voice detection based on software versions of this experiment running on the Macintosh and found such large response variances that the main effects were masked out, i.e., >200 msec. The voice detection software was written in MPW C using Apple Comnputer's Speech Recognition developer's application and a Macintosh microphone.

Word x Color patch, where the Word variable includes the color–only/no–word condition. There were 4 levels of Color (Red, Green, Blue, and Yellow) and 5 levels of Transparency (5%, 10%, 20%, 50%, 100%), which are described in detail in the section below. The 9 words were subdivided into 4 word-types: incongruent (4), neutral (3), congruent (1), and blank (1). This is consistent with traditional Stroop Effect experimental designs (e.g., MacLeod, 1991). A total of 180 trials resulted; 3 repetitions of this set were used, giving 540 trials in total per subject.

There were 4 Word x Color conditions (based on the standard Stroop categorization):

- incongruent color (a conflicting color word was presented e.g., for a color patch of red, words are: BLUE, GREEN, YELLOW),
- non-conflict or neutral (the word was a neutral word - NAIL, UNCLE, FOOD, CUTE appearing for each of the 4 possible colors),
- congruent color (the color word matched the color of the patch e.g., for a color patch of red, the word would be RED),
- and baseline (color only, no word appears).

An equal number of repeated measures of a baseline condition were included in the study: Transparency = 0% (opaque) Word type = none. Note this control condition is not orthogonal with Transparency, and so it must be treated separately from the factorial design described above.

The remaining 5 levels of transparency (5%, 10%, 20%, 50%, 100%) were applied across all words in the other 3 conditions (neutral, incongruent, congruent).

4.1.2.4 Stimuli

Stimuli were presented centered on the screen over a white background. Four colors were used: red, blue, green, and yellow. We purposely chose colors with widely varying hues and saturations. RGB values for these colors were: red (R=191, G=0, B=0), blue (R=0, G=127, B=255), green (R=0, G=176, B=0), and yellow (R=238, G=255, B=0). We initially tried an orange color (R=255, G=127, B=0), but pilot testing showed it to be too visually similar to the yellow. To minimize visual confusion due to indistinctive hue choices, blue was substituted. Pilot subjects could easily differentiate between these four colors. Much of the existing literature uses these four color choices, though none explicitly describe the RGB values (or other color properties) to allow exact matching or replication.

Four pilot subjects were run using the chosen colors with both variable length color patches and fixed length patches. The length of the color patch was determined based on either the word length of each stimuli (variable) or was determined based on the maximum word length possible (fixed). Using fixed or uniform sized color patches across all stimuli prevents inadvertent cues about the word or word length, which could possibly confound interference effects. However, this uniform patch size must be determined by the longest possible word. Short words would therefore have a high proportion of color area under the word. This introduced a potential problem with the ratio of color area to black text area, which would vary from one stimulus to the next (longer words fill the space while shorter words are centered within the space). To keep this ratio constant, variable length color patches would be required, which matched patch size to word size (length). We pilot tested both scenarios and determined that color naming performance produced similar results regardless of the patch size. We decided to use uniform patch sizes. This allowed us to use the exact same image files for our word naming experiment, where cues about word length are critical and therefore uniform size patches are required.

Words appeared through the colored rectangular patch (Figure 24). They were displayed in Helvetica, 78 point, uppercase. A sans serif font was chosen to minimize possible visual distortions to letters. A 78 point font size was chosen to ensure large, clearly visible stimuli from the subjects' seating position. Uppercase letters were used to create more uniform sized words. Words appeared in black typeface. Finally the ratio of black or dark text to color was maximized by choosing these large fonts and uppercase letters. All words were clearly visible when presented to the subjects at 100% transparency. We carefully selected neutral words UNCLE, NAIL, CUTE, and FOOD in addition to the four color names: RED, BLUE, GREEN, and YELLOW. Based on previous research (Klein, 1964; Dalrymple-Alford, 1972a) neutral words were chosen such that they did not convey possible confounding color meanings (e.g., sky - blue, grass - green, snow - white). Neutral words were also chosen such that they did not begin with a confounding color letter (R, B, G, Y) and such that they did not contain confounding endings (ED, UE, EEN, OW). Previous research has shown that even the first letter (Regan, 1978) or the word ending (Keele, 1972; Dalrymple-Alford, 1972b), if it is that of one of the colors used, results in some level of interference. All of the neutral words used have been applied in other studies of the Stroop test (e.g., Warren, 1972; Warren, 1974; McCain, 1983).

Transparency levels were varied as: 0% (baseline condition – only the color shows, the word is invisible), 5%, 10%, 20%, 50%, 100% (clear - both the word and color show) At 0%, the baseline condition is that of a simple color naming task with no interference possible. We purposely chose a skewed transparency distribution in order to minimize the number of overall trials, while maximizing trials in the region we felt to be most important. At levels between 50%

to 100% the word is easily visible and therefore high interference effects were expected. At lower levels, we were uncertain where the "break off" point might be. For this reason, more points were chosen between levels of 50% to 0%.

Images were produced using alpha blending (interpolated transparency, described in Foley, van Dam, Feiner, and Hughes, 1993, p. 764.) on a graphics computer (SGI Indigo™) to ensure accurate, exact transparency levels[3]. Using this technique, an exact combination of RGB from the foreground and background are combined to produce the resulting transparent appearance. (In effect, the alpha level controls how much of the back layer is allowed to "bleed through"). A pre-determined percentage of the background image is blended with the foreground image; the percentages of the background are reported in this dissertation. For example, "20%" means that 20% of the background was combined with 80% of the foreground to produce the resulting blended stimuli. Transparency levels were varied as follows: 0% (baseline condition – only the color foreground image shows, the word is invisible), 5%, 10%, 20%, 50%, 100% (clear - both the word and color show). At 0%, the baseline condition is that of a simple color naming task with no interference possible. We chose levels at which there were visually distinctive, noticeable differences (i.e., minimum of 10% differential for mid-range values). We purposely chose a skewed transparency distribution in order to minimize the number of overall trials, while maximizing trials in the region we felt to be most important. At levels between 50% and 100% the word is quite visible and therefore high interference effects were expected throughout this range. Since we anticipated small performance differences we tested only two levels in this range (i.e., 50% and 100%). At lower levels, we were uncertain where the "break off" point might be. For this reason, more points were chosen between levels of 0% to 50%. A total of 80 stimulus images were created in this way (4 colors x 9 words x 5 transparency levels). Pilot testing indicated that each of the resulting alpha blended (transparent) levels was visually distinctive. (A detailed discussion of the technical aspects of creating transparent images is provided in Appendix A. This appendix explains and compares several methods of implementation.)

---

3 This technique is far more accurate that the "faked" transparency typically achieved using masking techniques such as stippling or dithering (where alternate bits are simply turned off). Since there are a fixed number of bits available, many presumably different transparency levels appear visually the same with stippling; the values specified are rounded off to the nearest possible level (e.g., 15%=20%=25%=30% since the nearest possible value is 25%). The Macintosh supports stippling but not alpha blending. Images were therefore created with more accurate SGI graphics and were subsequently imported to the Macintosh which ran the experimental software. For our stimuli, each combination of word and color patch was produced at each desired transparency level (0%, 5%, 10%, 20%, 50%, 100%) as an 8-bit color TIFF file. Refer to Appendix A and B for more details.

Alpha blending algorithm:

$R_1 G_1 B_1 a_1$ current image in the frame buffer

$R_2 G_2 B_2 a_2$ overlapped image, a2 = some constant value for % transparency

$$R_3 = a_2 R_1 + (1-a_2) R_2$$
$$G_3 = a_2 G_1 + (1-a_2) G_2$$
$$B_3 = a_2 B_1 + (1-a_2) B_2$$

$R_3\ G_3\ B_3$ the resulting image combination

Where R_1, G_1, B_1 are the red, green, and blue color component values for the first image set pixels, R_2, G_2, B_2 are the red, green, and blue color component values for the second image set pixels, a_1 is the alpha value for the first image set, and a_2 is the alpha value for the second image set. The values for the combined, blended image are saved in $R_3\ G_3\ B_3$.

Each combination of word and color patch was produced at each possible transparency level (5%, 10%, 20%, 50%, 100%) as an 8-bit color TIFF file. These files were then converted on a Macintosh to 8-bit PICT resource files, in order to be used by the experimental software. A video card was installed on the Macintosh which allowed full 24-bit color displays. (More details about the image creation and conversion procedures are provided in Appendix A.)

4.1.2.5 Procedure

An instruction screen explaining the experimental procedure was presented to each subject prior to the start of the session (see Appendix B). Upon clicking a green "Go" button on the Button Box, subjects were then given 20 practice trials to familiarize them with the operation of the experiment. These trials were randomly selected from the set of 180 possible combinations. Subjects could ask questions if they wished immediately following the practice trials. Following this, again upon selecting the green "Go" button, subjects were shown 3 sets of the 180 combinations (15 minutes per set), with rest breaks in between each set. Trials were presented in random order at 5 second intervals. The 5 second inter-trial intervals were determined based on pilot testing. (These intervals were timed from the finish of a subject's response to the commencement of the next trial.) Each stimulus was presented for a maximum of 2 seconds or until the subject responded, whichever came first. Pilot testing showed that this provided ample time for a subject to respond and to prepare for the next trial. (This timing is consistent with other studies, e.g., Kahneman & Chajczyk, 1983.)

Subjects pressed the red "Error" button on the Button Box whenever they felt that they had made an error in response (i.e., whenever they were aware of saying an incorrect word). These errors were used as the experimental error rate for the various conditions.

We attempted to automate the error detection procedure through the creation of a speech recognition application. This was necessary since our experiment used voice responses. However, after three weeks of testing we determined that state-of-the-art speech recognition on the Macintosh was insufficiently accurate to be usable. Even with our small, restricted vocabulary and a highly directional headset microphone we could not obtain reliable error checking. Instead, subjects used a physical button press on the external Button Box during the inter-trial interval to indicate when they felt that they had made an error in response (i.e., whenever they were aware of saying an incorrect word). These errors were used as the experimental error rate for the various conditions. We justify this since previous studies have shown that subject logged errors do not differ from experimenter logged errors (e.g., MacLeod & Dunbar, 1988; MacLeod, 1994, personal communication). In addition, we analyzed videotapes from 4 randomly selected subjects to confirm this. The resulting numbers were completely consistent, confirming MacLeod's findings. Since subject logged errors were both accurate and efficient, we used this method to represent errors in responses. It is possible that using an explicit button press may have biased the subjects towards more cautious performance (speed versus accuracy trade-off). This was felt to be an acceptable trade-off.

The entire experiment took between 45 minutes to one hour per subject.

Subjects were debriefed at the end of the experiment. Open ended comments were recorded and the experiment was video and audio taped for analysis purposes. Response times and subject ranked errors were logged automatically by the software.

4.1.3 Results

Recall that our design used 4 word types: color incongruent word (3), color congruent word (1), neutral word (4), and no word (1), the latter of which equates to a transparency level of 0%. We used 5 additional levels of transparency: 5%, 10%, 20%, 50%, 100%. We had 4 colors: red, yellow, green, and blue. A 5 x 3 x 4 univariate ANOVA was carried out on the response time data with Transparency, Word Type, and Color as the independent variables. Note that Word Type only had 3 levels because the Baseline condition could not be meaningfully included in the ANOVA. This condition is not orthogonal with Transparency, and so comparisons involving it were conducted using separate F-test comparisons. Error trials were excluded from analysis and

are reported separately below. Post-hoc pairwise comparisons were conducted, here and throughout the dissertation, using a Student-Newman-Keuls (SNK) test with $\alpha = 0.05$.

Response Time Results

Transparency x Word type showed a significant interaction effect $F(12, 180)=4.18$, $p<.0005$ (Hypothesis 3). This interaction is particularly relevant to our original hypotheses (Figure 26). As hypothesized, significant main effects were found for transparency level, $F(4, 60)=6.23$, $p<.0005$ (Hypothesis 1), and word type, $F(3, 45)= 44.16$, $p < .0001$ (Hypothesis 2). Increases in transparency level clearly resulted in poorer performance (i.e., increased response time) (Figure 27a). Similarly, the choice of word significantly affected response times: incongruent or color conflicting words performed poorly, neutral and congruent words performed fairly well and equivalently, and color only (no word) was fastest (Figure 27b). Color also showed a significant main effect, $F(3, 45)=29.08$, $p < .0001$ (Figure 27c), indicating that saturation or luminance affect word legibility. There were no other significant effects.

The critical Transparency x Word interaction, shown in Figure 26, was investigated by conducting additional SNK tests of the effect of Word at each level of Transparency with $\alpha = 0.05$. Our primary interest was in the effect of Transparency under maximum interference conditions (i.e., Incongruent condition). At 5% transparency (word was only slightly visible), the Incongruent condition was not statistically different from the Baseline condition (color only - 0% transparency), indicating an absence of a Stroop effect. The words were not sufficiently legible to introduce any interference at these levels. The post hoc analysis also indicated that, at levels equal to and above 10% transparency (i.e., 10%, 20%, 50%, 100%), three groupings of means occur (as the Stroop effect would predict): baseline, congruent+neutral words, and incongruent words. (Separate post-hoc ANOVA and SNK analyses were carried out to explicitly compare the mean for the 0% condition against the other transparency levels.) Neutral and congruent word types show no significant differences across the entire range of transparency levels. This supports our expectation that transparency does indeed dilute the Stroop Effect (Hypothesis 3). The magnitude of interference for the Incongruent condition peaked at 50% transparency. – increasing levels of transparency did not further degrade performance. A post-hoc SNK analysis by word type showed that there were no statistically significant differences across transparency levels for congruent words or for neutral words (nor were these statistically different from each other) (Figure 26). (The high point at 20% for the neutral word case is partially attributable to a slightly elevated variance for this condition.) For incongruent words, response times for transparency levels occurred in five statistically significant groupings: 100% + 50%, 20%, 10%, 5%, an 0% (recall that 0% was the no–word condition) (Figure 26).

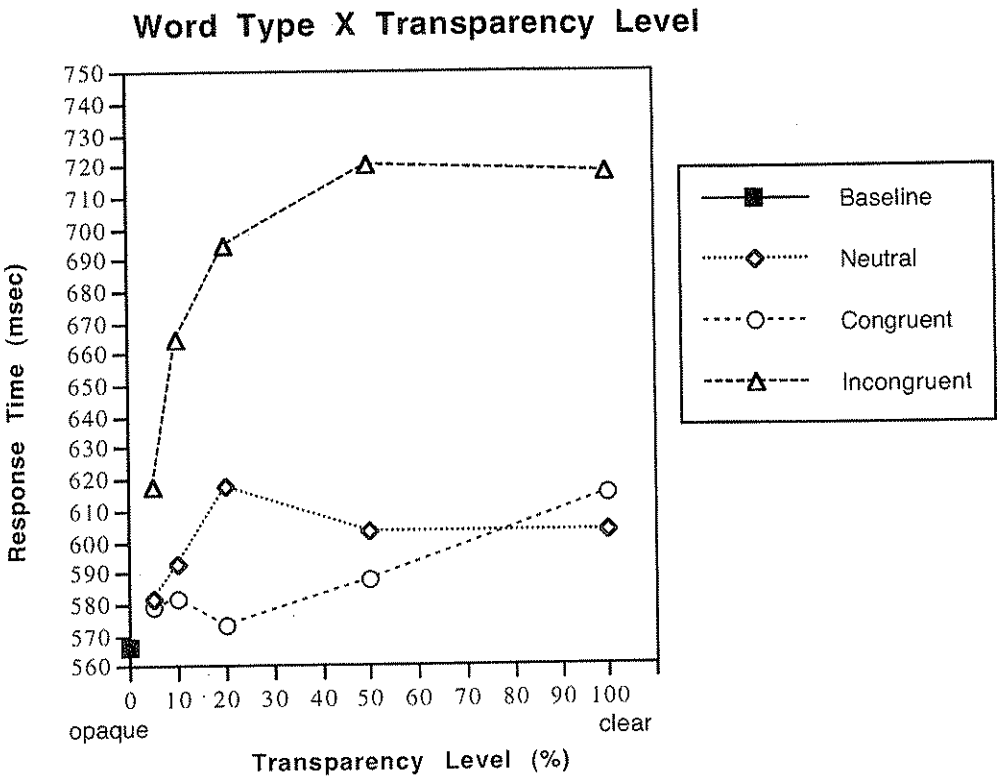


FIGURE 26. Word Type x Transparency Level Interaction - Color Naming Experiment

Post-hoc analyses (Student-Newman-Keuls test with $\alpha = 0.05$) were carried out to compare means for the main effects of Transparency and of Word. Response times for transparency levels occurred in three significantly different groupings (100%+ 50%+20%, 10%, and 5%) with higher levels of transparency leading to slower response times. The mean response times for each transparency condition are shown in Figure 27a. Note that the baseline condition is represented graphically as a single point. Recall that the baseline condition was the 0% transparent or opaque condition where no word appeared (color patch only).

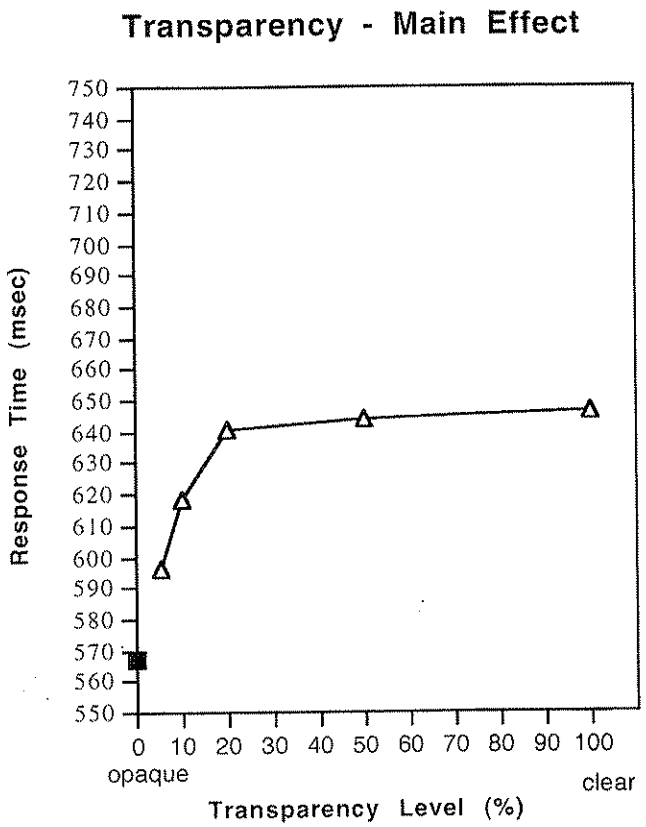


FIGURE 27. (a) Main Effects from Color Naming Experiment -Transparency Level

As expected, word types were statistically grouped according to the predicted Stroop Effect: incongruent (ink color conflicted with color word), neutral+congruent, and blank (color only - baseline condition) (Figure 27b). Some prior Stroop tests show significant differences between color congruent and neutral conditions (e.g., Regan, 1979; MacLeod, 1991). Our data did not pick up this distinction at a statistically significant level. However, our primary interest is not in the Stroop Effect per se, but rather in the relationship between transparency and interference (as reflected in the incongruent word condition).

A separate ANOVA was conducted to compare the baseline control condition (color only, no word) to the other Word type conditions. For this ANOVA, the control condition (no word) was treated as an additional level of the Word factor variable. The Transparency x Word x Color ANOVA was incremented to a 5 x 4 x 4 analysis (instead of the previous 5 x 3 x 4 analysis). A significant effect of Word was obtained $F(3, 45)= 34.74$, $p < .0001$. Post-hoc analyses showed that the Baseline control (color only) was significantly faster than the Incongruent condition and both the Neutral and Congruent conditions.

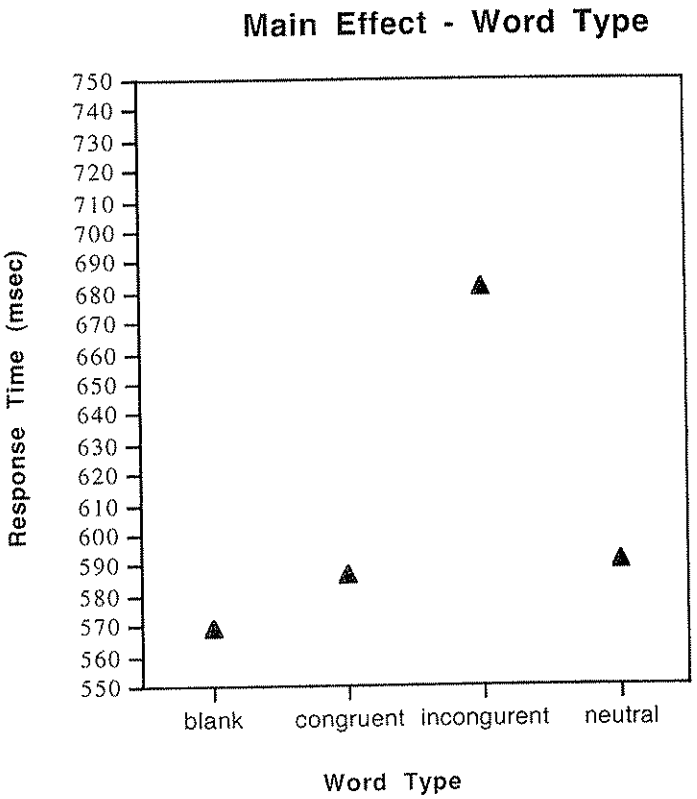


FIGURE 27. (b) Main Effects - Word

A post-hoc SNK analysis was carried out for the Color main effect. The green color performed significantly worse overall, followed by red and blue (about equivalent), and finally yellow (best). This suggests that, when using dark fonts, the luminance level of the colors has an effect on performance, with darker colors performing worse.

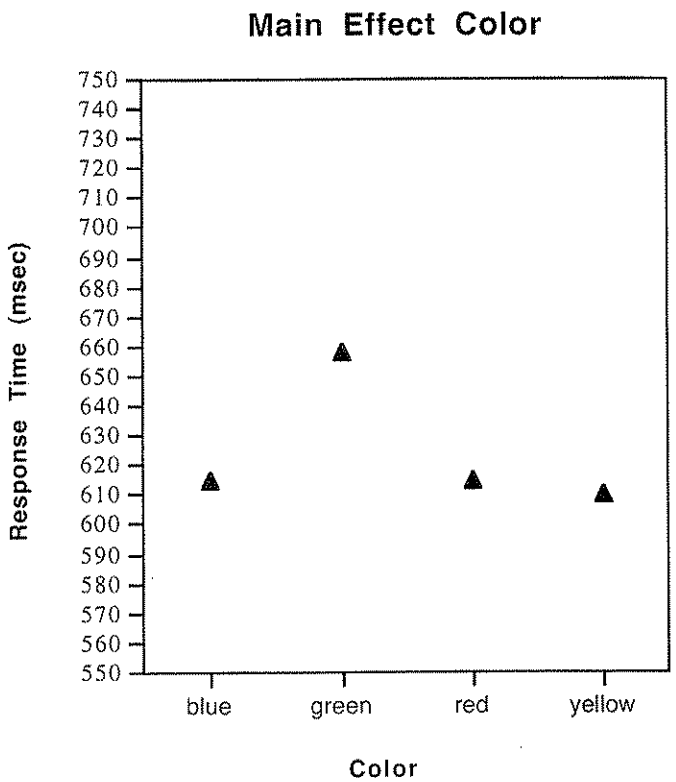


FIGURE 27. (c) Main Effects - Color

Error Results

Subject errors occurred only occasionally (average of 4 per 540 trials) and almost exclusively on the color-incongruent trials (Figure 28a). Analysis of errors by color shows red to be most error-prone (Figure 28b). The 5% transparency condition showed the fewest errors of all. Above this level of transparency, errors were approximately evenly distributed across levels (Figure 28c). This suggests that subjects were more careful under the more difficult, low transparency conditions. This further supports our belief that subjects avoided guessing responses. Note that levels below 20% (closer to opaque) show a pronounced drop-off in error rates. This primarily reflects the corresponding drop in interference from the background word (where interference invariably leads to higher errors).

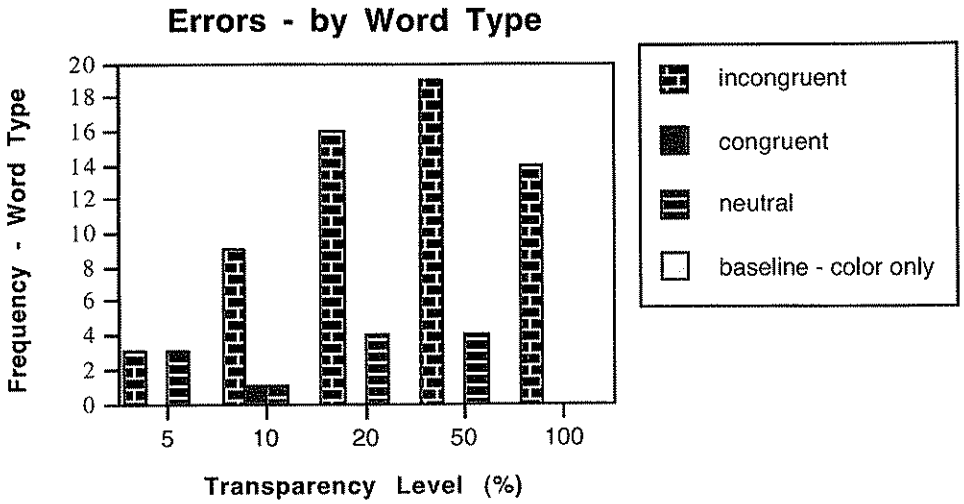


FIGURE 28(a). Errors in incorrect target identification

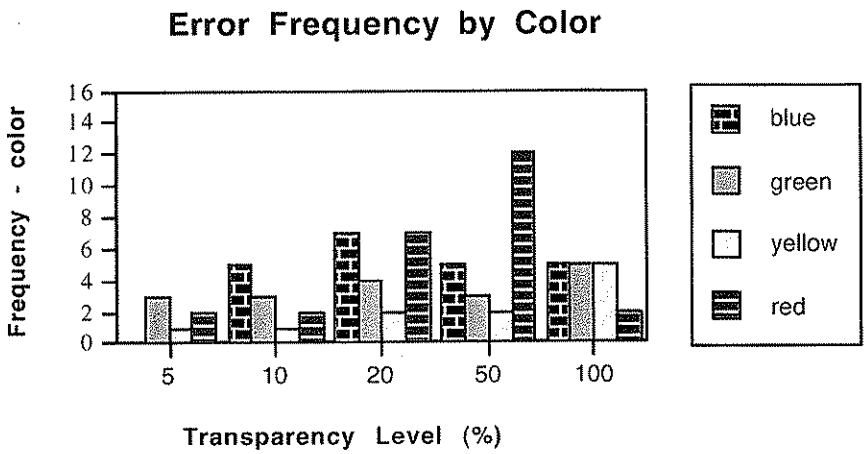


FIGURE 28(b). Errors classified by Color

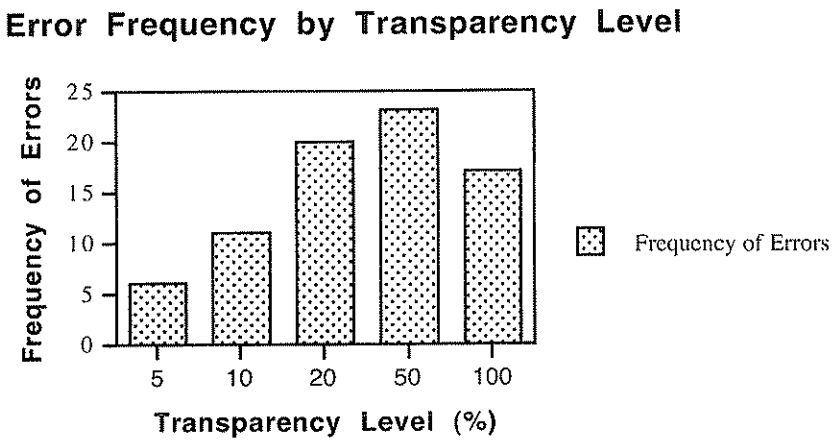


FIGURE 28(c). Errors classified by Transparency Level

4.1.4 Summary

We have found support for our hypothesis that increases in transparency result in increased interference, as measured using the color naming component of the Stroop tasks. Degree of transparency dilutes the interference/Stroop effect in a seemingly logarithmic fashion. In the color congruent case, performance levels off at 20% transparency, whereas in the color incongruent case, the leveling off point is at 50% transparency. It is this latter case, we are most interested in since it represents the most interference sensitive measure. Above levels of 50%, the word seems highly legible and performance does not seem to degrade further (i.e., interference is at a maximum). At levels of 5% (and likely less) minimal or no interference seems to take place (using SNK means comparison tests). At this point there is no performance distinction between different word types and hence there is no discernible Stroop Effect present. For this foreground focused attention task, this supports our proposed focus attention curve, with performance cut off points at 5% transparent (lower) and 50% transparent (upper).

It further seems that there was an interaction between saturation/luminance and legibility. In terms of response time, the green color performed worst overall, followed by red and blue (about equivalent), and finally yellow (best). This roughly implies that darker colors performed significantly worse than lighter colors, using dark fonts. In terms of error rates however, red was the worst, followed by blue, then green and yellow. The reason for this particular breakdown is difficult to determine. However, these results suggest that certain colors might be more profitably used in transparent windows or interfaces – though this remains to be tested. There is minimal Stroop literature in this area (MacLeod, 1991).

4.2 Experiment II - Word Naming

We used the word naming component of the Stroop paradigm to test the focused attention *background* curve (proposed in Figure 9). In this case, users are asked to ignore the color patch and read the word in the background layer. As in the previous Color Naming experiment, we measured both verbal response time and error rates. This experiment reflects a legibility test, necessary for background focused attention tasks (and potentially divided attention tasks where dual task visibility is required). The color patch in the foreground is always clearly visible. The transparency/opacity of the color patch was varied such that the word was sometimes clearly visible, sometimes faintly visible, and sometimes so faint that only the color appeared to show (there was always a word present however). By reading the background word, the user is, in effect, creating a background focused attention task, albeit a simplistic one. (In realistic HCI tasks, the user might be looking through a window or menu to see the object behind it.)

Sometimes the word was presented alone, with no color patch, as a baseline condition for word reading performance. (This replaced the opaque 0% color naming baseline condition.) At high levels of transparency (e.g., 100% - clear), it should be very easy to read the background word (focused attention on the background is easy). At lower levels of transparency (opacity increases), it should become progressively more difficult or impossible to read the word. Furthermore, we predicted that, at some transparency level between 20% and 0%, the word should be too difficult to read, and subjects should perceive that there is no word present. The results of this study should thereby reveal the parameters of the focused attention background curve in Figure 9.

The same Stroop stimuli were used to allow comparison of this experiment with the results of the Color Naming experiment. Results of previous research on the Stroop effect suggest that the word naming task should be significantly faster than the color naming task, and should be largely unaffected by the Stroop Effect (e.g., Stroop, 1935; MacLeod, 1991). Therefore, the primary purpose of this experiment was not to test for dilution of the Stroop effect due to transparency variations, but rather to test for word legibility and response time performance.

4.2.1 Hypotheses

H1: As transparency level increases (i.e., the word is more visible through the color patch) performance will improve (faster response time and fewer errors).

H2: There will be no difference in performance (error rates and response time will be equivalent) for word types within any given transparency levels, and hence no Stroop Effect.

4.2.2 Method

4.2.2.1 Subjects

The same 15 of 16 subjects participated in this experiment. (Note: one subject had to withdraw due to long term illness.) Order of presentation of experiments I and II was counterbalanced and presented at least one day apart to avoid possible transfer or learning effects.

4.2.2.2 Experimental System Configuration

The apparatus for this experiment was identical to that for the Color Naming Experiment.

4.2.2.3 Experimental Design

The independent variables were Transparency (5 levels), Word (9 levels), and Color (4 levels). Since this experiment consisted of a word naming task, we did not collapse the words into 3 categories, as in the Color Naming Experiment. Instead, each word was treated as separate level of the Word manipulation. We included the color names (RED, BLUE, GREEN, YELLOW), the neutral words (UNCLE, NAIL, CUTE, FOOD) and the baseline condition (word only, no color) as levels of the Word variable, hence the 9 levels. The dependent variables were verbal response time (measured in msec) and error in response.

A 5 x 9 x 4 (Transparency x Word x Color) fully randomized, within subject, repeated measures design was adopted. Color and Transparency were the same as in the Color Naming Experiment, except for the different control condition as described. We additionally conducted an ANOVA using the 4 word categories (congruent, incongruent, neutral, and word only) (i.e., a 5 x 4 x 4 analysis) with the results presented below. All subjects were exposed to all possible combinations of Transparency x Word x Color = 180 trials X 3 replications = 540 trials.

The order of presentation for each experiment was counter–balanced and spaced at least one day apart. Half the subjects did the Color Naming Experiment first, while the other half did the Word Naming Experiment first. (Each experiment could be treated as a within subjects blocking factor, i.e., the same subjects were used for both experiments, and the experiments themselves could be treated as 2 blocks presented in random order). No cross-task interference was anticipated based on previous research results (e.g., MacLeod and Dunbar, 1988; MacLeod, 1994), and indeed none was found.

The set of stimuli and conditions from the Color Naming Experiment were also used for Word Naming Experiment.

4.2.2.4 Stimuli

We used the exact set of stimuli from the Color Naming Experiment, with the following modifications: the 0% color only baseline condition (no word present) was replaced by a word only baseline condition (no color patch present). The word only condition was presented in the same font style and size as the other word–color conditions. Recall that color patches were a uniform size across all stimuli to prevent inadvertent cues about the word or word length, which could possibly confound interference effects.

4.2.2.4 Procedure

The procedure for this experiment was identical to that of the Color Naming Experiment described earlier, with the exceptions described below.

Subjects used a button press (the red "Error" button) during the inter-trial interval to indicate an error in response. In the word naming task, previous results indicate that subjects make very few errors. (Our results also showed that errors were very seldom made so the workload of this activity was minimal.)

Subjects were told that there would always be a word presented however, sometimes it would be very difficult to read this word. They were asked to try their best to read the word but if they could not see any word present (i.e., appeared to be color alone), they were to respond "none" and press a yellow button. This ensured that for all trials, subjects were consistently required to verbally respond. (Conscious suppression of a verbal response turned out to be much more difficult and error prone. Subjects had little difficulty getting used to saying "none" if they could see no word and pressing a button.) This does increase the risk that subjects might reduce their efforts to read the word and simply respond none much more frequently. However, analysis of both pilot data and experimental results (based on the instructions we gave to subjects) indicated that subjects seemed highly motivated to attempt the trials (i.e., very low numbers of error responses were logged across all subjects).

One concern was potential for confusion since subjects being required to press one button for "wrong word spoken" and a second button for "no word". This did not turn out to be a problem for several reasons. First, errors in response occurred so rarely that subjects were not forced to frequently make button press choices. Second, subjects were told that there was no response time "deadlines" for pressing the button and the inter-trial interval time duration turned out to be ample for the required decision making about which button to press (or whether to press one at all). Third, since there were only two possible button press choices, subjects kept one hand on each button and quickly seemed to develop an association between saying "none" and using their right hand (over the rightmost button) versus saying an incorrect response and using their left hand (over the leftmost button). Subjects commented that they found this two-handed strategy worked well.

Both subject logged errors and trials marked "none" were removed from response time analysis and were subsequently analyzed separately. In rare instances, we noticed that a trial was indicated as being illegible from the "none" button press, yet the transparency levels were such that a word must have been quite visible (e.g., 50% or 100%). These were taken to be response

errors. In this way, we would over estimate the error rates slightly in order to be overly cautious and conservative in our subsequent analysis.

Each experimental session lasted about 45 minutes in total, 15 minutes per each set of 180 trials. Rest breaks were inserted at 15 minute intervals.

4.2.3 Results

A 5 x 9 x 4 univariate repeated measures ANOVA was carried out on the response time data with Transparency, Word, and Color as the independent variables. Error trials were excluded from analysis and are reported separately below. We subsequently conducted an analysis collapsing the Word variable into categories of: Neutral, Congruent, Incongruent, and None, and carried out a 5 x 4 x 4 ANOVA. These results are also reported below.

Response Time Results

There was a significant interaction between Transparency x Color $F(12, 163)=6.17$, $p < .0001$ (Figure 29). This suggests that word legibility is affected not only by level of transparency (i.e., visibility) but also by the properties of the color used (i.e., saturation and luminance). As hypothesized, a significant main effect was found for Transparency, $F(4, 56)=34.31$, $p< 0.0001$ (Hypothesis 1) (Figure 30a). Transparency increases resulted in performance improvements (reduced response times), as hypothesized. Color and Word (Hypothesis 2) also showed significant main effects: Color, $F(3, 42)=52.85$, $p < 0.0001$, and Word, $F(8, 98)=6.25$, $p < 0.005$ (Figures 30b and 30c). No other significant interactions were observed.

A separate ANOVA (5 x 4 x 4), based on word type categories, was conducted to compare the means for the Word effect: $F(4, 56) = 6.25$, $p < .005$. Post-hoc tests showed that the Baseline condition (word only) was not statistically different from any other Word type at the 100% transparency (i.e., no Stroop effect present). The 0% baseline condition is graphically represented as a single point, since there can be no transparency differences without a color patch. While word type showed a significant main effect in response times, this was not ordered according to a Stroop Effect; in other words there was no Stroop Effect as hypothesized. Also there were no interaction effects between the Word variable and any other factors.

Post-hoc analyses were carried out to compare means for the Transparency main effect (Student-Newman-Keuls test with $\alpha = 0.05$). Transparency levels occurred in three significant groupings: 5%, 10%, 100%+50%+20%+0% (across all word types). Recall that 0% was the word only baseline, which was not statistically different from the 100% (clear) condition (word with color

background). As illustrated in Figure 29, response time improves with increasing transparency, as expected.

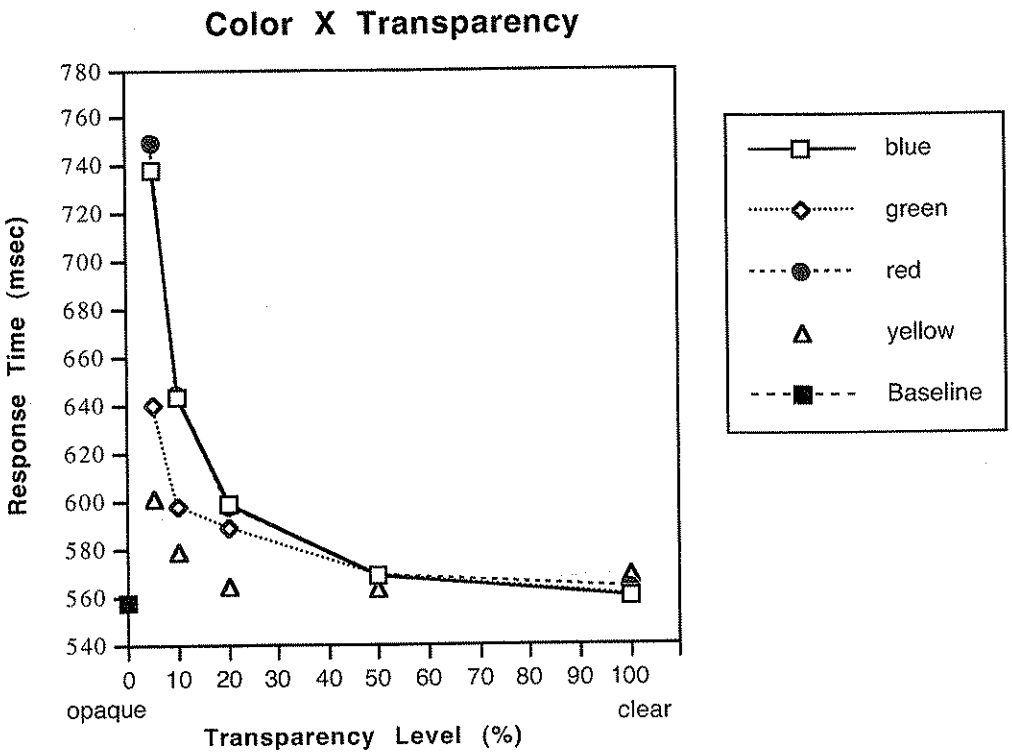


FIGURE 29. Color X Transparency Level Interaction - Word Naming Experiment

Post hoc analysis of Color (Student-Newman-Keuls test with $\alpha = 0.05$) revealed statistically significant groupings of: yellow, green, and red + blue (Figure 30b). Overall, yellow was easiest (i.e., fastest response times) followed by green. Blue and red were the slowest, but not statistically different from each other. (In Figure 29, red and blue curves are virtually overlapping.) These color orderings reflect the luminance level orderings (yellow is highest, green was mid-way, red and blue are lowest). Luminance levels appear to have the most dramatic effect at low levels of transparency when visibility is most impacted. At levels over 50%, color properties cease to differentiate performance since all stimuli are highly visible regardless of hue, saturation, or luminance levels. This interaction effect is not completely surprising given the high contrast between our black font versus the light (i.e., yellow) foreground.

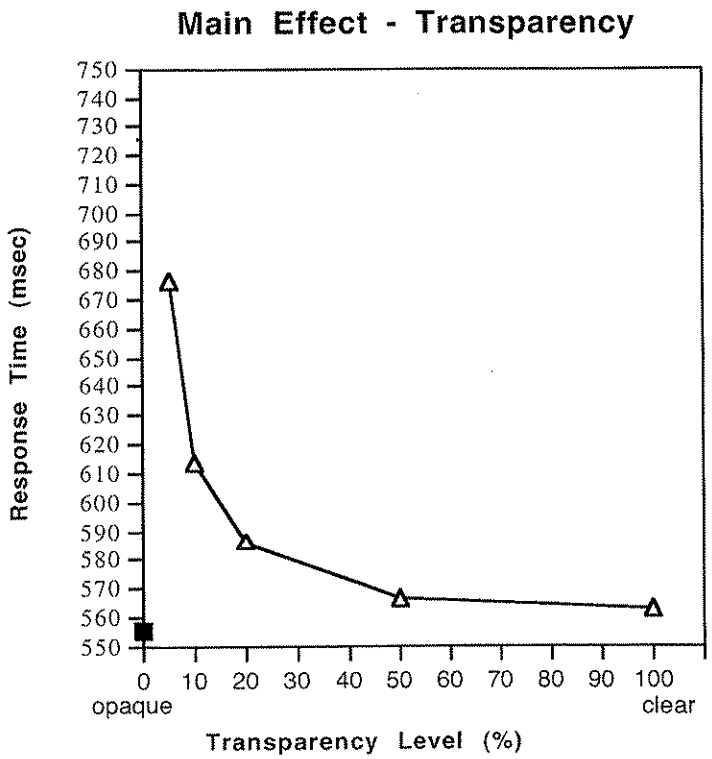


FIGURE 30 (a). Main Effects - Transparency Level

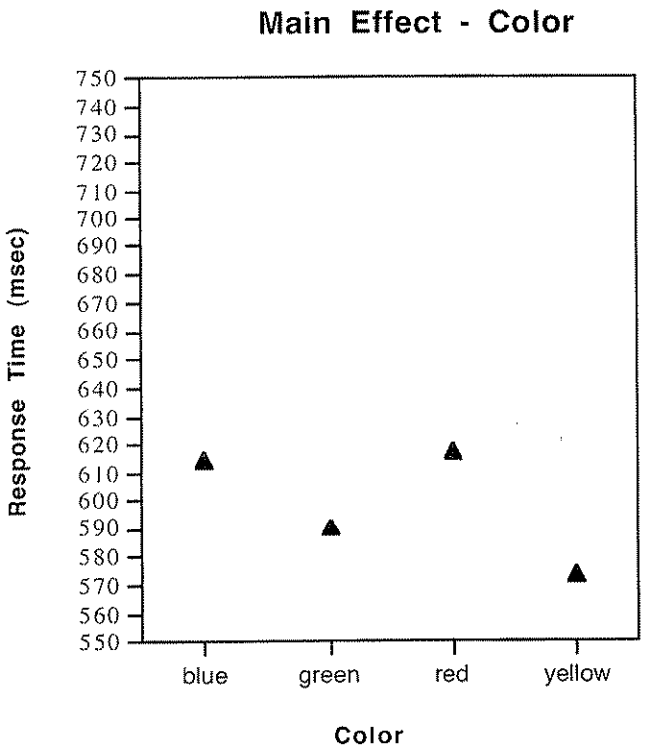


FIGURE 30 (b). Main Effects - Color

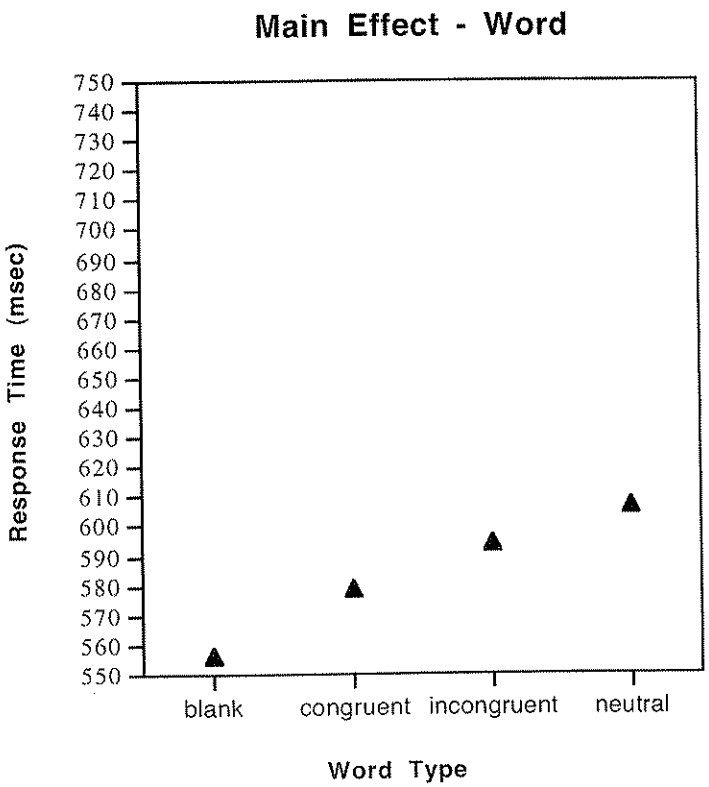


FIGURE 30 (c). Main Effects - Word Type

Error Results

Trials where the subject named the wrong item or the color instead of the word (incorrect response) were marked as errors. These trials tended to be very few and most often occurred in the low visibility condition (5% transparency) (Figure 31a). Furthermore, these trials occurred under what we now believe to be the most difficult visibility condition (red or blue colors). It is also possible that incorrect naming behavior resulted from either momentary distraction or from slight interference caused by the immediately preceding trials, according to several comments subjects made during and after the experiment.

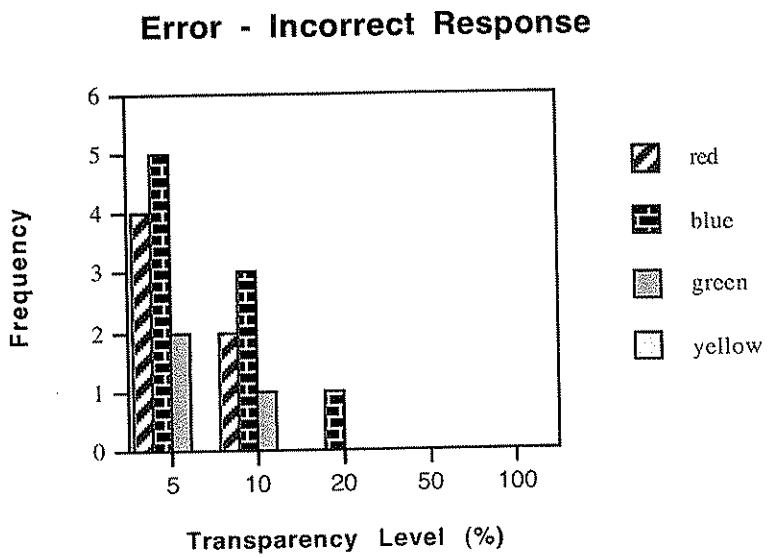


FIGURE 31. (a) Errors - wrong word named

Error analysis of *illegible* trials (those subjects marked "can't see") also supports our belief that luminance level interacts with legibility. Low visibility conditions (i.e., transparency = 5%) are the most highly sensitive to interactions with color properties (Figure 31b). At lower levels, subjects reported great difficulty in seeing the word. This was more predominant for the darker, more saturated colors. About 15% of these trials were errors. At 5% and 10% levels, certain colors produced better performance than others. For transparency levels above 10%, subjects made virtually no errors and performance was consistent across colors. At 20% levels and higher, all words were easily read and, based on a post-hoc SNK analysis of color, there were no significant differences in response times. Yellow was the most robust color in terms of overall legibility, even under conditions of low transparency (i.e., poor visibility).

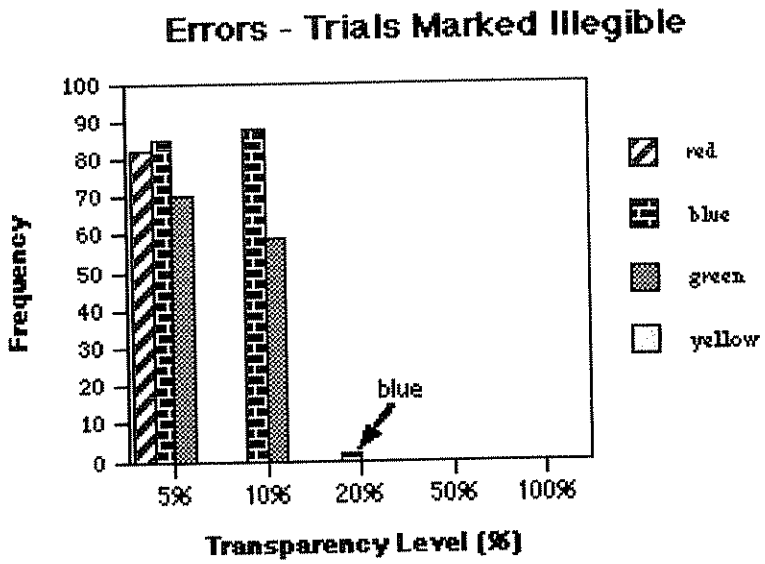


FIGURE 31. (b) Errors - Trials marked "Can't See" by subjects

4.2.4 Summary

We have found that degree of transparency had a direct effect on word legibility. Word naming performance improved dramatically, with performance leveling off at 20% and not improving beyond that point. At levels of 5% (and likely less), word naming is highly error prone and almost unusable. At levels of 10% subjects could accurately name most of the words, though they performed slightly better with some background colors. As hypothesized, there is no performance distinction between different word types, and hence, there is no discernible Stroop Effect present. In general these results support our proposed background focus attention curve. However, we underestimated the effect of increased transparency. As a consequence the proposed performance cut off points are at 10% transparency (lower) and 20% transparency (upper). We believe that this rapid performance improvement and peaking is closely related to the complexity of the task, in particular the visual complexity of the imagery. In our experiment, we used 78 point Helvetica fonts and short, simple words. This was necessary to properly conduct a balanced Stroop Experiment. However, this generated an exceedingly simple word–reading task, which turned out to be fairly "interference resistant" (where interference is visual interference resulting from transparency). As a result, we hypothesize that a more difficult task (e.g., one which has finer lines and details) would produce a more gradual curve and would shift the peak to a higher level of transparency. The predicted low end leveling off point did not occur. We believe that this was because users chose the "cannot see any word" option when the effort to read the word became excessive. At 10% transparency the number of trials marked illegible steeply increased, implying that this level represented a user perceived "threshold of frustration". Beyond this point, users often marked trials as illegible rather then exerting additional effort to read the word. This represented reading efforts which typically took more than 1000 msec or 1 second to respond to.

Furthermore, there was an interaction between saturation/luminance and legibility. In terms of both response time and illegible trial errors, the red and blue colors performed equally poorly overall, followed by green and finally yellow (best). Again, this suggests that darker colors performed significantly worse than lighter colors, when using dark fonts.

4.3 Experiment III - Legibility and Word Shape

In Experiment II, several of the subjects commented in the post-experiment interviews, that they were uncertain whether they had really been able to read certain low-visibility words. They felt that they could see shadows of the letters and based on the length of the word (and the fact that there were a small number of words to recall), they guessed what the correct word was. This led us to believe that the task was too simple and that results might have been confounded by this shadowing effect. We could find no prior literature on this for the word naming experiment. We wanted to test the true legibility of the words (reading performance) rather than educated guesses. To this end, we re-created a new set of words which matched existing words in both letter shape and length. We recalled 5 subjects and replicated the word naming experiment using these new stimuli. We anticipated that, if subjects truly used word shapes to make inadvertent guesses about words, the low transparency level conditions would result in response time groupings based on these word pairs. Note that we could not have used these same words in a color naming task since the endings (e.g., REEN, ED) would confound interference results (e.g., GREEN, RED), based on previous findings from others (Keele, 1972; Dalrymple-Alford, 1972b). To avoid confounding the color naming Stroop Effect, we only used these new words in a word naming experiment.

4.3.1 Hypothesis

H1: Performance (errors and response time) within words pairs (e.g., BLUE and CLUE, RED and LED) will be equivalent but performance across "word pairs" will show a statistical difference. (This would reflect the fact that subjects were using properties of the shapes of words, as they initially believed, to facilitate reading performance.)

4.3.2 Method

4.3.2.1 Subjects

A total of 5 students from the University of Toronto were recalled from prior Experiments (within subjects design).

4.3.2.2 Experimental System Configuration

The apparatus for this experiment was identical to that for Experiment I and II.

4.3.2.3 Experimental Design

The independent variables were Transparency (5 levels), Word (9 levels), and Color (4 levels). Since this experiment consisted of a word naming task, each word was treated as separate level of the Word manipulation. We included the color names (RED, BLUE, GREEN, YELLOW), the neutral words (LED, CLUE, PREEN, FELLOW) and the baseline condition (word only, no color) as levels of the Word variable, hence the 9 levels. The dependent variables were verbal response time (measured in msec) and error in response.

A 5 x 9 x 4 (Transparency x Word x Color) fully randomized, within subject, repeated measures design was adopted. Color and Transparency were the same as in the Color Naming and previous Word Naming Experiment. We additionally conducted an ANOVA using the 4 word categories (congruent, incongruent, neutral, and word only) (i.e., a 5 x 4 x 4 analysis) with the results presented below. This gave us the same 9 x 4 x 5 = 180 trials x 3 replications = 540 trials.

To facilitate comparison, a within subject design was used for this experiment by recalling subjects from the previous experiments. Since this experiment was conducted to investigate possible confounding factors derived from the Word Naming Experiment, order of experimental presentation could obviously not be counter-balanced. However, we did not anticipate any cross-task interference since this experiment was conducted several months after the others.

4.3.2.4 Stimuli

We used the same set of stimuli from Experiment II, with several key modifications. The color words remained unchanged: RED, BLUE, GREEN, and YELLOW. The neutral words were now altered to be: LED, CLUE, PREEN, and FELLOW. Note that since the word naming experiment is generally insensitive to the Stroop Effect and to the problem of semantic interference, we do not anticipate any differences in word naming performance based on word type. Furthermore, in Experiment II, we did not obtain a Stroop Effect with *exact* color words, therefore we do not anticipate any confounding problems by using the endings of color words.

The same colors and uniform patch sizes were used as in Experiment II.

As previously described, words appeared "through" the colored rectangular patch, displayed in Helvetica, 78 point, uppercase, black ink.

As in Experiment II, transparency levels were varied as: 0% (baseline condition – only the word shows, there is no color patch present), 5%, 10%, 20%, 50%, 100% (clear - both the word and

color show). At 0%, the baseline condition is that of a simple word naming or reading task with no interference possible.

4.3.2.5 Procedure

This was the same procedure used in Experiment I and II.

4.3.3 Results

A 5 x 9 x 4 univariate repeated measures ANOVA was carried out on the response time data with Transparency, Word, and Color as the independent variables. Error trials were excluded from analysis and are reported separately below. We subsequently conducted an analysis collapsing the Word variable into categories of: Neutral, Congruent, Incongruent, and None, and carried out a 5 x 4 x 4 ANOVA. These results are also reported below. Error trials were excluded from analysis and are reported separately below.

Response Time Results

There were no statistically significant differences between the results of the previous Word Naming Experiment and the results of this experiment (Figure 32), based on a pairwise comparison of means at each transparency level and an SNK comparison of means. This suggests that the shadowing effect which might have assisted guessing under low visibility conditions did not occur. (Figure 32 shows overall mean response times comparing each of these two experiments.) Consistent with the previous Word Naming experiment, there was a significant interaction between Transparency x Color $F(12, 48)=10.74$, $p < .0001$. Additionally there was a significant interaction between Transparency x Word $F(32, 128)=2.83$, $p< .0001$ (which did not appear in the prior Word Naming Experiment). This suggests that word legibility is affected by not only the properties of the color used (i.e., saturation and luminance), but also by level of transparency (i.e., visibility), in cases where the words are very similar. A significant main effect was found for transparency, $F(4, 16)=25.58$, $p< 0.0001$. Word type and color also showed significant main effects: Word type, $F(8, 32)=21.69$, $p < 0.0001$, and Color, $F(3, 12)=81.47$, $p < 0.0001$.

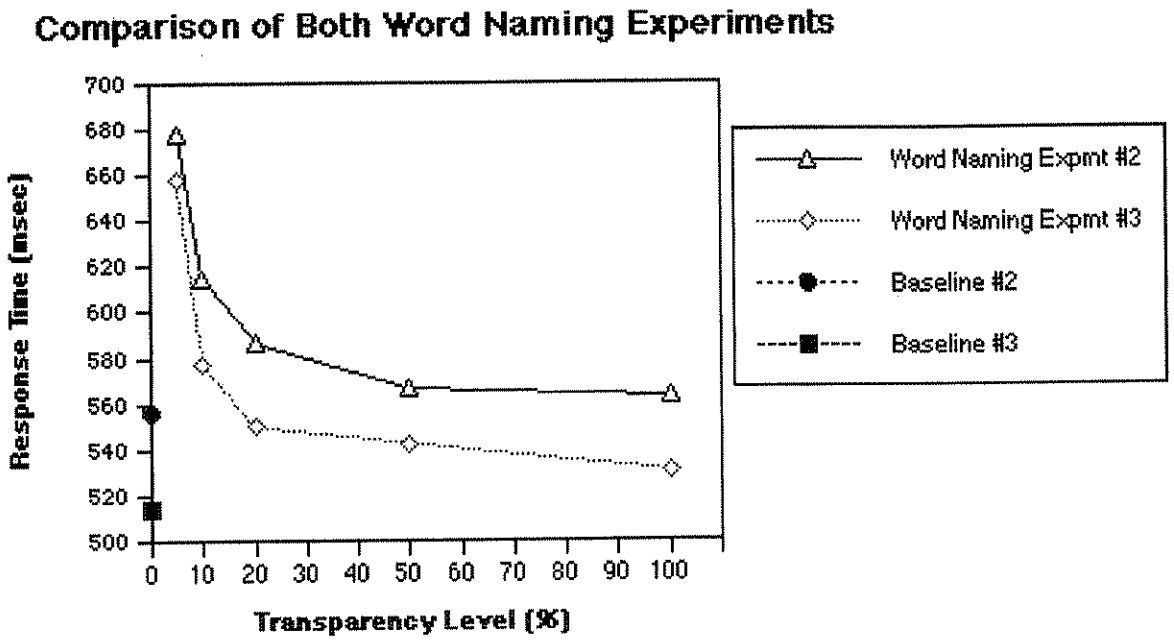


FIGURE 32. Graph of Results from Experiment III - Word Naming

Post-hoc analyses were carried out to compare means for Experiment II versus Experiment III results, for word types, and for transparency levels (Student-Newman-Keuls test with $\alpha = 0.05$). Post hoc analysis of color X transparency level interaction effect (Student-Newman-Keuls test with $\alpha = 0.05$) revealed statistically significant groupings of: yellow, green, and red + blue (as found previously). Anaylyses of the main factor effects showed that Transparency levels occurred in the same three significant groupings as previously noted: 5%, 10%, 100%+50%+20%. Post-hoc analysis of transparency level across both Experiment II and Experiment III (Figure 32) (i.e., pairwise comparisons), revealed that there were no statistically significant differences at any level, between the two experiments (which does not support our hypothesis). We can find no evidence to support the "assisted guessing" hypothesis, as it might relate to word shape or letters. Furthermore, an SNK analysis by word type did not reveal the pairwise differences we would expect if our hypothesis held.

Error Results

The five subjects recalled for this experiment made no errors in response, nor were any trials marked illegible. We believe that the more difficult matched words used in this experiment resulted in subjects increasing their attention for each trial and, as a consequence, they made no errors. There may additionally have been an inadvertent effect of using only neutral words. After a time, subjects would realize that no color conflicting words appeared. Therefore, overall the trials were slightly easier.

4.3.4 Summary

We found no evidence that our choice of words or the characteristic shapes of the letters provided inadvertent cues for the word naming task. When words of visually similar character shapes to the color names were employed, no significant or unanticipated differences in performance were found. Furthermore, results from word naming tasks using such words confirmed out earlier findings. Color and Transparency main effects and signficant clustering of means remained the consistent with the previous Word Naming experiment.

4.4 Discussion

The Stroop test was used to evaluate interference between transparent layers because it provides a sensitive, robust measure of the extent of interference. As such, it suggests worst case limitations. When combined, the Color Naming experiment (foreground focused attention) and the Word Naming experiment (background focused attention) should indicate the trade-off points and curves for attentional performance and legibility.

From Experiment I on Color Naming, we learned that degree of transparency dilutes the Stroop effect in a logarithmic fashion, with performance leveling off at between 20% and 50% transparency. Above 50% transparency, there is little additional interference; it is already at its maximum. This interference is reflected both in response times and error rates. At levels at or below 5%, minimal or no interference seems to take place and interference effects are no longer detectable. This is reflected in both the faster response times and the lower error rates. These results support the focus attention foreground curve proposed in Chapter 2 – Figure 9 with performance cut off points at 10% (lower) and 20% (upper). We additionally observed that color interacts with the extent of interference. We attribute this to the saturation/luminance levels of the different colors and their affect on word legibility.

From Experiment II (Word Naming Experiment), we learned that word naming is highly error prone at levels of 5% (i.e., unusable). At levels of 10%, subjects could accurately name most of the words, though they seemed to perform slightly better, depending upon what the background color was. Yellow was best, followed by green, then red and blue. It seems clear that there was an interaction between saturation/luminance and legibility. Yellow was most interference resistant (i.e., it supported much more opaque levels without affecting word legibility as severely).

The combined response time between the Color Naming and Word Naming Experiments are presented below (Figure 33). If we were to determine a possible "optimal performance point" for

both tasks, it would lie at the point of intersection of the two curves. Note that for these two tasks, the graph seems to indicate a direct trade-off between foreground and background focused attention. We would estimate this trade-off point between 10% and 20% transparency. Note that this theoretical optimum point is specific to this particular set of tasks and assumes that neither task has an inherent priority. Both tasks are weighted equally in contributing to the determination of this optimum. In reality, this assumption may not be valid since often one task has at least a momentary priority over another (e.g., selecting an item from a foreground menu and then continuing a background drawing task).

While the Stroop Test reflects *semantic* interference, through the design of this experiment we have factored in a *visual* interference component. Semantic interference results from a mismatch between a required verbal response and the *meaning* conveyed by the printed word. On the other hand, visual interference results from conditions where one item reduces *legibility* of another item. Visual interference can result from the occlusion of one item by another, the overlapping of information without occlusion (e.g., subtitles on films or sporting events which are partially transparent but nevertheless reduce visibility on the bottom of the picture), or overlapping matched types of information (e.g., color on color, text on text, contour lines over contour lines). Our Color Naming and Word Naming experiments combine both types of interference. The Stroop test we used is based on semantic interference. However, our experimental design also generated a visual interference condition through the use of transparency. Words which cannot be seen due to the opacity of the color patch (i.e., visual interference) cannot generate semantic interference. In this way, we created conditions under which the semantic interference is related directly to the level of visual interference (visibility due to transparency/opacity). This will allow us to apply our results to other scenarios comprised exclusively of visual interference conditions. It is primarily this visual interference (resulting from transparency) which plays a key role in more general user interface design problems.

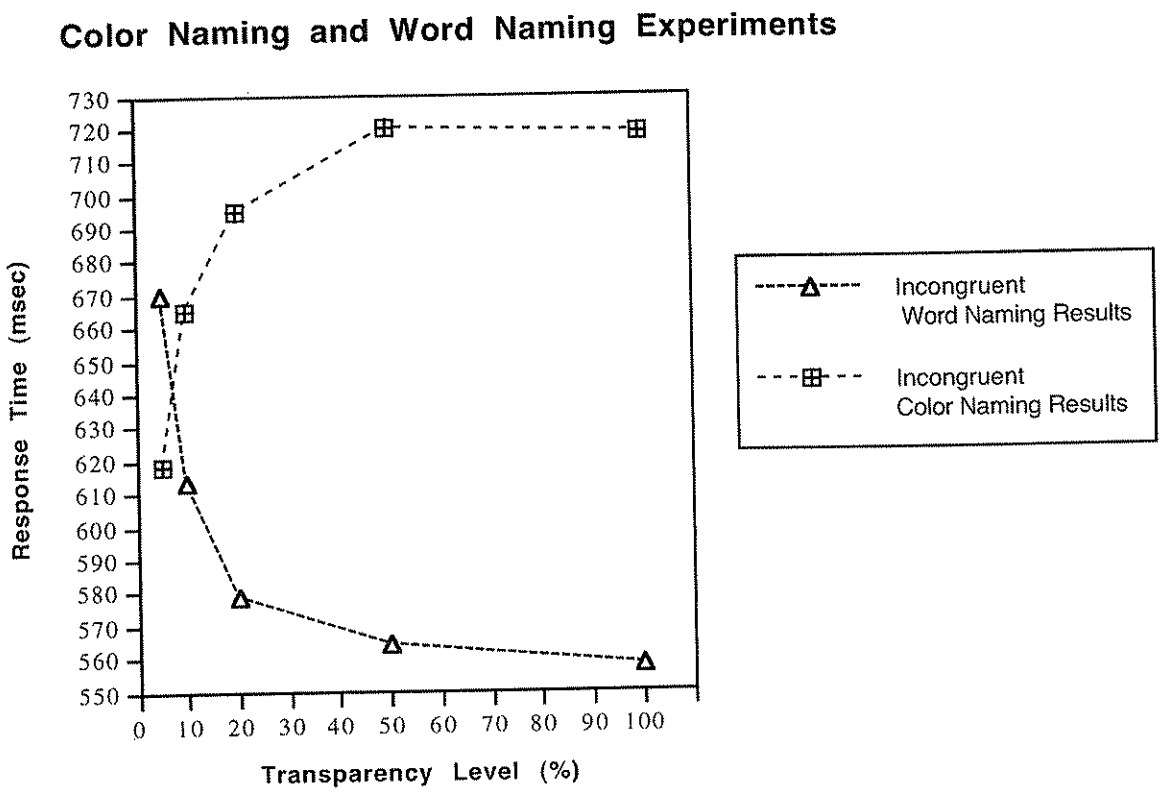


FIGURE 33. Comparison of Foreground and Background Task Performance

4.5 Implications for Design

These experiments presented a novel use of the traditional Stroop paradigm by varying the degree of word transparency, thereby linking visual interference to semantic interference, and observing the effects on the degree of task interference. The main purpose for doing so was to obtain a conservative estimate of the optimal trade-off between supporting focused attention on the foreground and focused attention on the background. This represents one of the few studies on legibility and the Stroop Effect. In addition, this experiment uniquely examines the crossover of word legibility and word interference. Subsequent chapters will generalize this methodology and apply these results in more representative tests of transparent human-computer interfaces.

The Stroop test was used to evaluate interference between transparent layers because it provides a sensitive, extreme measure of the extent of interference. As such, it should suggest worst case limitations. (Note that in real world tasks, the interference caused by the background object is less likely to have observable performance degradation at very low visibility levels.) Our results suggest that for background focused attention tasks, substantial performance gains occur within the first 20% transparency levels, but may not occur from 20% to 100%. Levels of 5% or less do not seem usable.

For foreground focused attention tasks, there is a rapid performance degradation between 5% and 20% transparency. At 20%, performance is at its worst and does not deteriorate substantially with further increases in transparency. Our foreground attention curve represents a highly performance–sensitive task and as such should over-estimate performance degradation with transparency effects. Combining these results, and factoring in error rates, we expect an optimal performance point between levels of 10% and 20% transparency.

Different tasks will have different levels of error tolerance and acceptable performance limits, based on the consequences and costs of making errors and the cost and likelihood of error recovery. Also, the legibility of multiple layers will be determined by the visual distinctiveness of the stimuli in addition to overall transparency levels. The Stroop Effect experiments tested interference and visibility at a very simplistic level using a simple task. We have learned a number of useful guidelines for predicting possible performance curves.

Visually simple tasks or tasks with visually dissimilar information seem more interference resistant and hence can be expected to generate performance curves which are flatter and are shifted towards faster response times. The number of errors in such tasks should also be lower than for more visually complex tasks. There will be a lower and upper threshold of performance for both foreground and background focused attention tasks, after which further changes in transparency level will have no significant performance impact. Legibility is at an optimum when at least 50% of the image is visible (i.e., 50% transparency or higher). However, further increases in legibility do little to further improve performance. In the case of highly disparate information, such as that used in the Stroop Experiment (a word versus a solid color patch), 20% transparency works as well as 50% transparency. Reducing transparency to 10% or less effectively eliminates interference from a second information source. However, the acceptable lower thresholds of performance are likely to be strongly influenced by what is an acceptable level of errors and what the cost of these errors is or the cost of recovery from them would be. Finally, the "threshold of frustration" is a critical determinant in the error rate or the point at which subjects are no longer willing to expend additional effort. This is affected by the cost of errors, subject motivation, fatigue, etc. This value can be determined based on an analysis of the response times for those trials where subjects chose the "illegible trial" or "can't see" option and the transparency levels where the number of such errors increased.

4.6 Summary

This chapter looked at assessing legibility and visual interference using the Stroop paradigm, with transparency as an independent variable. By using this conservative and robust test of visual interference, we have been able to assess the shape and parameters of the performance curves for both a foreground and a background focused attention task. The shape of the resulting curves puts constraints on a viable theory of psychological mechanisms behind the Stroop effect, as it relates directly to variations in transparency level. To date, this particular manipulation i.e., transparency and legibility, has not been reported in the literature. Of primary interest to us, we have related the concept of visual interference and legibility (through transparency variations) to semantic interference, as reflected by the color naming component of the Stroop experiment. Also, formerly neglected in the existing literature, we have applied a rigorous specification of the colors chosen. It is our hope that additions to this body of literature will carry on this tradition, greatly facilitating replication of results.

The next stage of our research is to determine if these results generalize to more complex and realistic tasks. Also, we need to better understand the effect of transparency on visually complex information which is reflective of the possible target domains that originally motivated this work. This is addressed in the experiments described in Chapter 5 and the prototypes described in Chapter 6.

5.0 Transparent Tool Palettes and Menus

The series of experiments in this chapter represents the progression from tightly-controlled, theoretically motivated experiments to empirical studies which sacrifice some experimental control in order to increase representativeness (and thus applicability to everyday tasks). Our objective is to better understand how transparency might affect user performance in a standard graphical user interface (GUI). We broadly define standard GUIs to be systems which consist of overlapping windows or dialogs, where users must select objects (most often using a mouse) from the top-most window. The objects selected include text labels, graphical icons, sliders, buttons, or other bordered areas. Usual performance measures in GUI systems are selection accuracy of a particular object and speed of selection (or response time). We can further define accuracy in two ways: a bulls-eye target where the center point is a "direct hit" and the magnitude of the error increases with distance, or a simple hit-miss strategy applicable for targets with precise boundaries (such as sliders or buttons). Most often GUI objects fall into this latter category since they have well-defined visual boundaries; this is especially true of items typically presented in foreground windows such as menus, tool palettes, and interactive dialogs. Our foreground task experiments therefore consider targets with well-defined borders where errors are of the hit-miss type and selection response time is measured as the time it takes to locate, move to, and click on the specified item using a mouse. We test both graphical object selection performance and text selection performance since these are the two most common general elements in window design. The second major component of interest to us is a "click-through" background selection task, which exist in only a few advanced systems such as the ToolGlass system (Bier, et al., 1993; Bier, et al, 1994; Stone, et al., 1994) and the CaveDraw system (Lu and Mantei, 1990). In this case, users are selecting either a particular feature of an object or a small graphical object in its entirety (e.g., an icon) from the background, through an overlapping see-through foreground window or palette. We have chosen object feature selection and the bulls-eye error type for our background task experiment, which reflects realistic task characteristics. In all cases, we varied the level of transparency between the foreground and background, and we measured the effect this had on performance. This provided us with a measure which relates transparency to user performance for both foreground item selection (applicable to menus, windows, and palettes) and for background selection (indicating feature

identification ability). By varying the type of information contained in the foreground and background, we also explore the impact information type has on interference which will ultimately affect performance.

We used the first series of experiments, reported in Chapter 4 to inform our experimental design and the hypotheses for this series of experiments. The derivations and connections between experiments are described in detail below. The same basic experimental methodology was used with several notable modifications. In the place of the Stroop color patch, which appeared as our previous focused foreground attention task, we now insert either an iconic tool palette or a linear text menu. We replace verbal response times with mouse click selection times. Finally, we replace our 78-point Helvetica word from the previous Stroop experiments (background layer), with complex background images taken from product libraries. These images reflect a "snapshot" in time for several task domains we have targeted for later case study evaluation. They also represent the task domains used in our static prototype evaluation. While the images used as stimuli are not interactive, they do reflect a static moment in time for our choice of tasks. As in the Stroop experiment series, we conducted both focused foreground attention tasks and focused background attention tasks. These tasks have been carefully matched to allow comparison.

5.1 Experiment IV - Iconic Tool Palette Selection Task[1]

For this experiment, we used a variably-transparent tool palette superimposed over different background content files. Both the icons in the palette and the background content were visually complex. The objects on the tool palette were taken from existing icons used in a real product, although these icons were not generally used together on the same palette in the real application. The palette contained text icons, line art icons and solid, rendered object icons. This choice allowed us to evaluate content-based interference problems but meant we combined disparate icons from various tool palettes within the product. The background content contained text pages, wire-frame images, and solid images. The background content was also selected from a set of real working images contained in the product library as released to customers. The palette was presented at varying levels of transparency. We evaluated both the effect of varying transparency levels (from opaque palettes to highly-transparent palettes), and the interference

1 A summary of this experiment was published as Harrison, B. L. Kurtenbach, G., and Vicente, K. (1995). An Experimental Evaluation of Transparent User Interface Tools and Information Content. In Proceedings of User Interface Software and Technologies UIST'95, November 14-17, 1995, Pittsburgh, Penn. 81-90.

produced by different types of content information (e.g., Figures 34, 35a, 35b). The experiment did not test product usability or icon purpose. The use of icon palettes over library images does reflect realistic usage of UI tools for this product. Subjects were not very familiar with the actual working product. However, we do not anticipate that the discrepancies from the real application would invalidate our results, given the goals of the experiment and the background of the subjects.

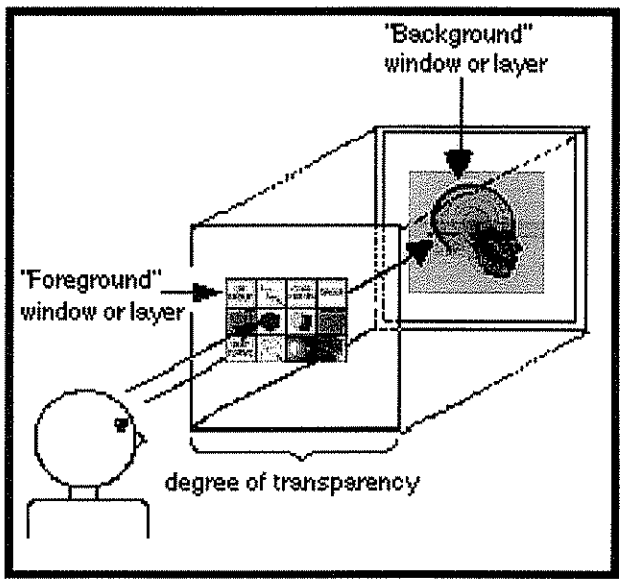


FIGURE 34. Experimental Overview (Modified Stroop test figure)

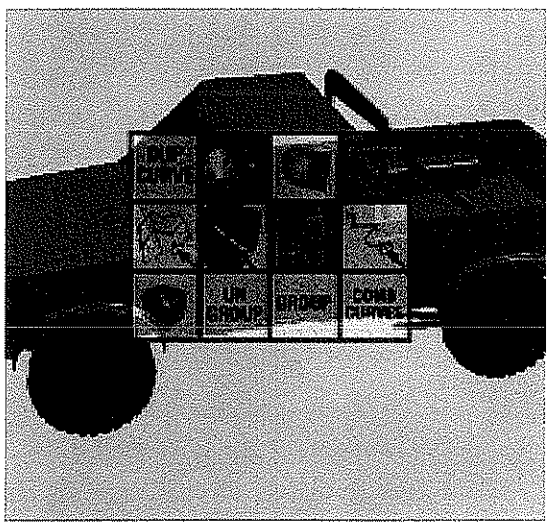
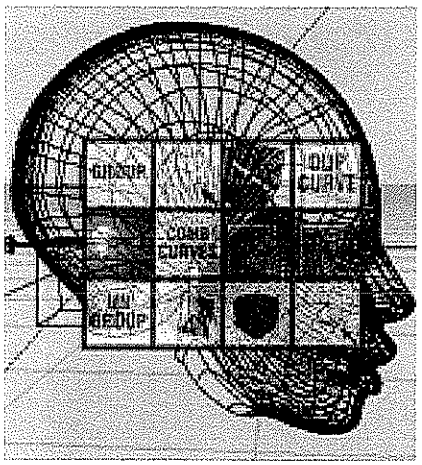


FIGURE 35 (a) Sample Palette over Solid Image.



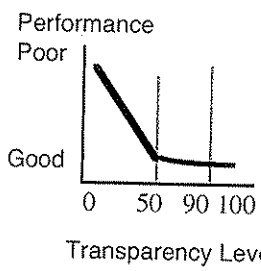
(b) Sample Palette over Wire Image

5.1.1 Predictions Based On Previous Experiments

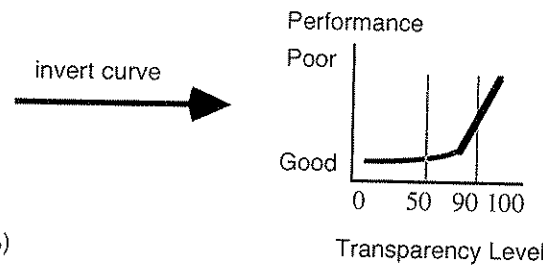
The Palette Selection experiment reflects components of both Stroop tasks. It represents a focused attention *foreground* task and, as such, we anticipate a performance curve which

resembles those depicted in Figures 26 and 27 (i.e., opaque foregrounds should have "good/fast performance" and transparency increases should degrade performance). However, unlike the Stroop color naming experiment where the colors were unambiguous, in the palette selection task specific features must be used to determine the correct target from visually similar items (i.e., the task is more similar to a legibility task such as the Stroop word naming experiment). Performance is unaffected by semantic interference but is expected to be sensitive to changes in visibility (i.e., sensitive to visual interference). Also, in the color naming experiment, the color patch appeared over the word but was still clearly visible *around* the word at 100% transparency. In the case of designing transparent icon palettes, the icon image, itself, must be made transparent since the image fills the palette space. If we use the 100% transparent or clear condition, it reflects a no-icon condition (i.e., a hole in the palette). To draw a parallel with the color naming experiment, the Stroop word itself would have had to appear in a colored ink which gradually became transparent and did not show outside the font boundaries. These differences suggest that the predicted performance curve for the icon palette task should be modified to resemble a word naming experiment, but inverted to be a foreground attention task (shown in Figure 36).

Our previous Stroop word naming results showed little performance difference from 50% transparent to the "best reading condition" (where "best" is 100% transparent in the case of a background task and presumably 0% or opaque in the case of a foreground task). This suggests that the resulting palette selection task might achieve best performance at 50% with little significant improvement between 50% and "best reading condition" (0% or opaque). Furthermore, in our Stroop word naming task, levels below 10% were found to be error prone or illegible (defined as 10% of the background word and 90% of the foreground color patch forming the displayed image). If we again translate this value to our palette selection foreground task (inverting the task), 90% transparency may be a cut-off point, where 90% transparency roughly means that 90% of the background image and 10% of the foreground palette image are used to create the displayed image.



36 (a) Word Naming Experiment curve, shown here as a background task



36 (b) Projected curve for the Palette Selection Experiment

FIGURE 36. Predicted Performance Curve

The second curve and cut-off points shown above (Figure 36b) represent our prediction of results, prior to conducting the experiment. We have carefully outlined the method and reasoning used to derive this curve. We can now state the hypotheses based on this performance prediction.

5.1.2 Hypotheses

H1: As transparency level increases (i.e., the background is more visible through the icon palette), performance will be poorer as reflected by increases in errors and slower response time (interference will be higher).

H2: The content of the background image (text, wire frame, solid) will have a large interaction effect with legibility of the icons, as transparency level increases.

We expect two effects. First, we anticipate that increased complexity or information density on the background will make icon legibility decrease for *all icon types* (i.e., a main effect for background type). Text backgrounds will have the worst performance, followed by wire-frame, then solid images. Second, we also anticipate that icon types which are visually similar to background types in terms of both color and content will be most affected in terms of performance degradation (i.e., text icons with text background, line art icons with wire frame backgrounds, and solid image icons with solid image backgrounds).

H3: Lastly, we expect to observe proposed cut off points at least 50% (lower bound) and at most 90% (upper bound), reflected in the data (based on our prior experimental results).

5.1.3 Method

5.1.3.1 Subjects

A total of 13 students from the University of Toronto participated as subjects. They were pre-screened for color vision deficiencies and for familiarity with the product from which the images and icons were taken. Subjects were paid for their participation and could voluntarily withdraw without penalty at any time.

5.1.3.2 Experimental System Configuration

The experiments were run on an SGI Indy™ using a 20 inch color monitor. Subjects sat at a fixed distance of 60 cm from the screen (average distance when working normally).

5.1.3.3 Experimental Design

A 12 x 12 x 4 fully randomized, within subject, repeated measures design was used. There were three independent variables: Icon type (3 levels), Background type (3 levels), and Transparency (4 levels). The types of icons were: solid, line art, and text, with 4 unique samples of each type (12 combinations). The types of backgrounds were: solid images, wire frame models, and text pages, with 4 unique samples of each (12 combinations). The 4 transparency levels were: 0%, 50%, 75%, 90%. Transparency levels were based on our previous experimental experience and test pilot results with this experiment. The opaque level (0%) represented the baseline condition where the fastest performance was anticipated. Pilot results suggested minimal performance improvements between 0% (opaque) and 50% (semi-transparent) so intermediate levels within this range were not included. Comparative analysis of 0% to 50% was sufficient to determine whether our prediction about a 50% cut-off point was correct. Similarly, images above 90% transparency were found to be almost completely illegible and were therefore not included. A 100% (clear) condition was not included since this implies the palettes, by the nature of the transparent design, would be completely clear and hence unidentifiable. A total of 576 trials were used for each subject.

5.1.3.4 Stimuli

We used three icon types: text, line art, and solid rendered objects (Figure 37). Within each of these three types, we selected four samples from existing product icon palettes. Our resulting tool palette was 3 rows by 4 columns in size. A 12 item palette was felt to be representative of the average menu/palette size used within the target application (i.e., the range of items per palettes was found to be from 6 - minimum to 14 maximum, with a median size of 12). Icons were randomly assigned positions within the palette for each trial. While this is clearly an unrepresentative choice, it is more consistent with the focus of our experiment, which was legibility (and not learning). Having a fixed menu order would be more representative but it would allow participants to learn the position of different menu items, and thereby perform the task based on memory, not legibility. Randomizing the order of items on the menu thereby provided an unconfounded estimate of legibility performance. The target was presented to the

subject throughout the trial as a reminder. This was to prevent errors due to memory (which we were also not testing for).

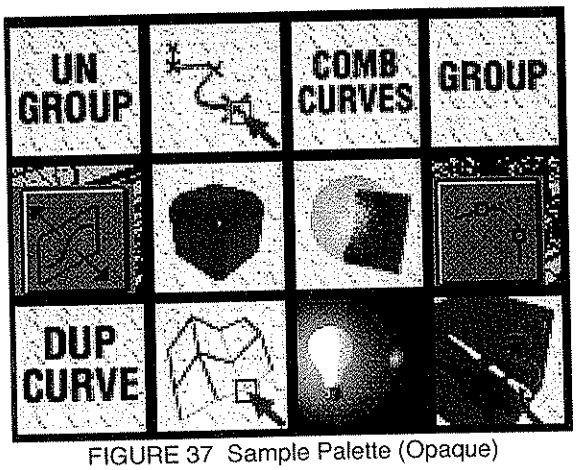


FIGURE 37 Sample Palette (Opaque)

Text icons were selected in pairs based on matching similar letters and words, while still using real product icons. These similarities were deliberate to ensure a stringent legibility requirement. Likewise, similar pairs of line art icons were selected. Finally, for solid icons, we chose items that were frequently used in the product, representing a variety of both shape and color components.

We created background images of three types: text pages, wire frame images, and solid images (e.g., Figure 38, 39, 40). Four samples of text pages and of each image type were created. The text pages contained different layouts in terms of paragraph formatting and different fonts (Times, Helvetica, Geneva, and Courier were used). Wire frame images were 8-bit colored images taken from the product library. Both dense and sparse wire frame images were included. Also the predominant color in the wire frames varied from blue to red. Solid images were 8-bit color rendered. Images were chosen such that a variety of colors and complexity was represented. Some images had numerous multi-colored parts, whereas others were predominantly a single color, such as red (truck) or black (camcorder). For each image, we generated three slightly different views or orientations so that the alignment of background to icon palette was randomized. All backgrounds were displayed such that a major portion of the content was aligned directly under the palette.

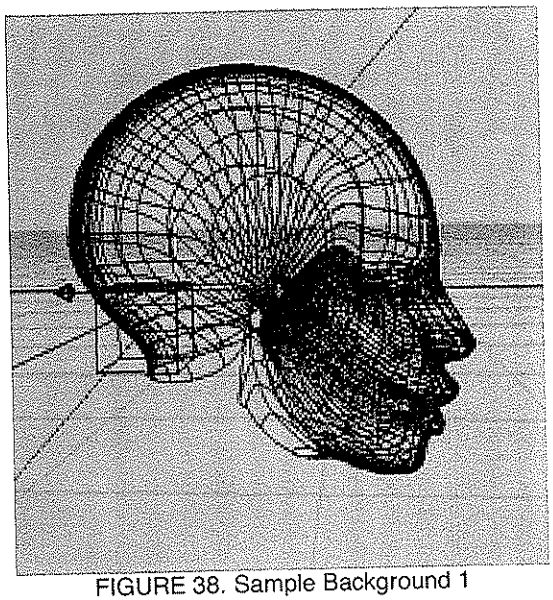


FIGURE 38. Sample Background 1

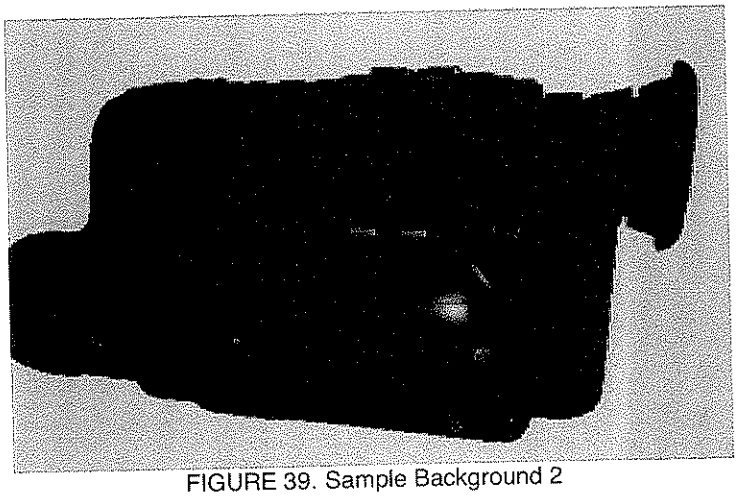


FIGURE 39. Sample Background 2

were three independent variables: type of font, type of background, and transparency level. A total of 720 trials were run for each subject. Trials were presented in random order. Each session lasted about 45 minutes. Dependent variables of subjects response time (based on a mouse click) and errors were logged. Two error conditions were possible: the subject pressed the "can't see" button indicating that the item was not legible, or the subject selected the incorrect menu item. In the latter case, the item selected and its location were logged. Error trials were removed from subsequent analysis of response time data. Error data was analyzed separately. We used 2 groups of menu items: each group was visually similar to ensure true legibility performance. The menu items were: Revolve X, Revolve Y, Revolve Z, and Dep Curve, Comb Curve, Del Curve. Six other menu items were randomly distributed with the target items. (A 12 item menu was felt to be representative of the average menu time menu size used within the actual product.) Items were randomly assigned positions within the menu for each trial. This was done to ensure the experiment was not confounded by subjects learning the position of items. The target item was presented to the subject throughout the trial as a reminder. This was to prevent memory errors (which we were not testing for). We randomly assigned background images of three types: text pages, wire frame images, and solid images. Again four samples of each type were created. Images were 8-bit color rendered images. These backgrounds were aligned such that a major portion of the content was directly under the menu.

FIGURE 40. Sample Background 3
Note that all background images were reduced substantially for printing, resulting in poor image resolution in this medium. Images were high resolution for the experiment.

To produce the exact image combinations required, we used alpha blending techniques as described in Chapter 4. This ensures precise combinations for any specified level, without graphical artifacts. For a more detailed discussion and comparison of the possible methods to produce transparency effects, refer to Appendix A.

5.1.3.5 Procedure

The subjects' task was to locate and click on a particular target item in the iconpalette. The target icon was presented to subjects throughout the trial at the bottom left of the screen (lower left in Figure 41), so that they would not have to rely on their memory. Again, this is consistent with the objective of obtaining an unconfounded measure of legibility performance. To prevent guessing, if the item on the palette was not legible, subjects were told to press a "can't see" button located at the bottom right of the screen (lower right in Figure 41).

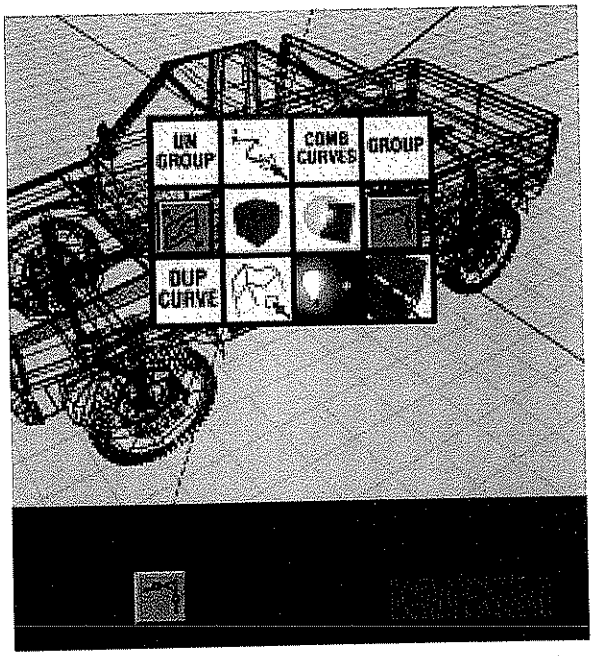


FIGURE 37. Sample Trial Screen

Subjects were given brief verbal instructions followed by 20 practice trials. These trials were randomly selected with replacement from the set of 576 possible combinations. Subjects could ask questions if they wished during, or immediately after, the practice trials. For each trial, subjects were shown a randomly selected target icon for each trial and a button to start each trial ("Next Trial" button). This gave subjects time to study the target icon prior to commencing each trial. The timer started as soon as this button was pressed and the button press brought up a randomly chosen background. Subjects had to locate and click on the target item within the palette. The timer stopped as soon as a palette item or the "Can't See" button was selected.

Subjects could take short rest breaks whenever necessary and were debriefed at the end of the experiment.

Trials were presented in random order. The target icon was randomly selected from the group, with replacement. Icons were randomly assigned a position in the palette for every trial. We randomly assigned background images. Finally, we randomly assigned the level of transparency to the palette. All 12 x 12 x 4 possible combinations were used. A total of 576 trial were conducted for each subject. Each session lasted about 45 minutes (one session per subject).

Selection response times and errors were logged. Response selections were made using the mouse by clicking on either a palette item or by clicking on the "Can't See" button. Two error conditions were possible: the subject pressed the "Can't See" button indicating that the item was not legible, or the subject selected the incorrect palette item. In the latter case, the incorrect item selected and its location were logged. Error trials were removed from subsequent analysis of response time data and were analyzed and reported separately. At the end of the experiment subjects were asked which items they felt to be "easiest" and "hardest" and if they were aware of employing any particular strategy. The experimenter additionally noted any observable strategies based on movement patterns. Subjects' responses and any open ended comments were recorded and are reported below.

5.1.4 Results

We have categorized our results by Response Time analysis, Error analysis, and comments from the interviews with subjects. A 12 x 12 x 4 univariate ANOVA was carried out on the selection response time data with Icon type, Background type, and Transparency as the independent variables. We used a conservative cut-off point for our ANOVA – alpha level = .01. The 4 samples of each Icon type and of each Background type were later collapsed for a subsequent 3 x 3 x 4 ANOVA, reported below.

<u>Quantitative Statistical Analysis - Response Time</u>

The experimental results indicate a significant interaction effect ($p < .0001$) was found for: Icon x Transparency (Figure 42), Background x Transparency (Figure 43), and Icon x Background. A significant 3-way interaction was found for Icon x Bakcground x Transparency. Highly, statistically significant ($p < .0001$) main effects were also found for Transparency (Figure 44a), Icon (Figure 44b), and Background (Figure 44c) However, we are primarily interested in the interaction effects with Transparency. The statistical results are summarized in Table 1 below.

condition	df	F value	p<
icon	11, 132	8.40	.0001
background	11, 132	6.94	.0001
transparency	3, 36	38.02	.0001
icon X transp	33, 394	6.15	.0001
bkgrnd X transp	33, 368	8.70	.0001
icon X bkgrnd	121, 1452	2.72	.0001
icon X bkgrnd X trans	299, 2834	5.72	.0001

TABLE 1. Statistical Results for Main Effects and Interactions

To determine if the differences are significant between the individual lines plotted within each of the graphs, a Student-Newman-Keuls (SNK) test was run post-hoc as a comparison of means, with an alpha level = 0.05. (This determines the clustering of items within font type, background type, and transparency level and indicates which items are not statistically different.) The results of the post-hoc analyses are described below.

To understand the 3-way interaction effect, it is easier to first present the data on the 2-way interactions and subsequently discusse the Icon x Background x Transparency interaction. We first examine the Icon x Transparency interaction in more detail.

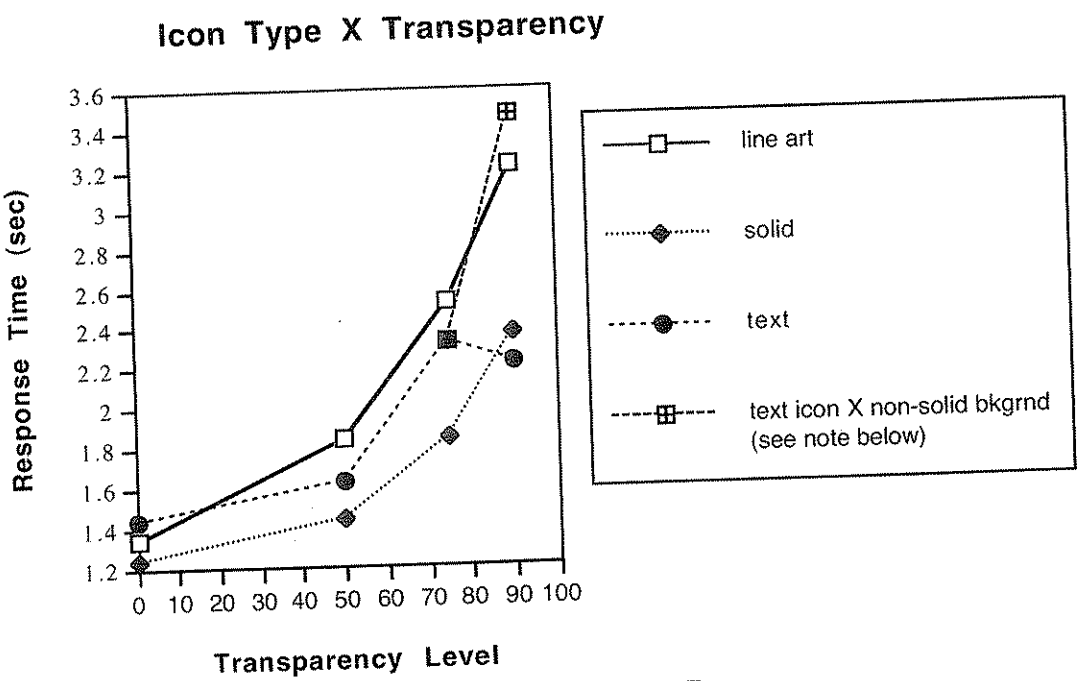


FIGURE 42. Mean Response Times for Transparency Levels X Icon Type (across all background types)

A post-hoc SNK analysis of the Icon x Transparency interaction revealed that all levels *within* any given Icon type showed statistically significant differences (i.e., the levels within any

individual line plotted), excepting the 90% text icon (Figure 42). With each increase in Transparency, a statistically significant performance improvement over the preceding level was observed. In the case of the 90% Transparency X text icon point, when taken as text icons over *solid* images, this point is not different from solid icons. When restricted to text over *non-solid* backgrounds (the extrapolated point shown in Figure 42), this point is not significantly different from the line art icon performance at 90% transparency. However, as we anticipated, for 0% through 50% transparent, there were no statistically significant performance differences *across* Icon types, implying that at fairly opaque levels the icon is highly visible over the background and the type of icon plays an insignificant role. At 75%, solid icons performed significantly better than text icons or line art icons (which were not significantly different). At 90%, there is a significant difference between line art icons and solid icons. The ranking of text icons depends upon our interpretation of the data (see comment below). Text icons over solid image backgrounds are the predominant completed trials and are comparable to solid icon type performance. The few trials attempted with text over anything except solid backgrounds are comparable to line art icon performance.

Upon examining more closely the text icons x Background interaction, an explanation for these results can be postulated. Note that the *text* icons at 90% showed a surprising *decrease* in response time. This is primarily due to the nature of the trials attempted by subjects at this level. At the 90% level, most text targets included in the above analysis were text icons over *solid* image backgrounds, which seemed easier (i.e., faster) to determine. Subjects determined that text icons over *non-solid* backgrounds at the 90% condition were too difficult and therefore most of these trial conditions were marked "can't see". The effect of this was to include fewer overall trials at 90%, where most of these trials had faster response times (i.e., a skewed distribution). (We subsequently normalized these data to account for the skewed distribution by applying log(RT) .) The response time results for *non-solid* backgrounds at 90% are plotted as a special symbol in Figure 42 for comparison purposes. These non-skewed, normalized results are as we might have predicted.

To determine if the differences are significant between the transparency levels plotted within each background type (i.e., Background X Transparency interaction), a Student-Newman-Keuls (SNK) test was conducted. Wire frame backgrounds (with any icon type) showed no statistical differences in performance between 50% and 0% (opaque) (Figure 43, performance leveling off point). Solid backgrounds (with any icon type) showed no statistical performance difference between 75% and 0% (opaque) (Figure 43, performance leveling off point). Text backgrounds showed no significant differences from 0% to 50% transparency. These results imply that icons were highly visible up to 50% transparency regardless of the background type although, as we

anticipated, solid backgrounds performed slightly better than text pages or wire frame models (which were not different from each other). At 75%, solid backgrounds performed significantly better than text pages or wire frame images (which were not significantly different). At 90% (highly transparent), all background types performed equally poorly (no significant differences).

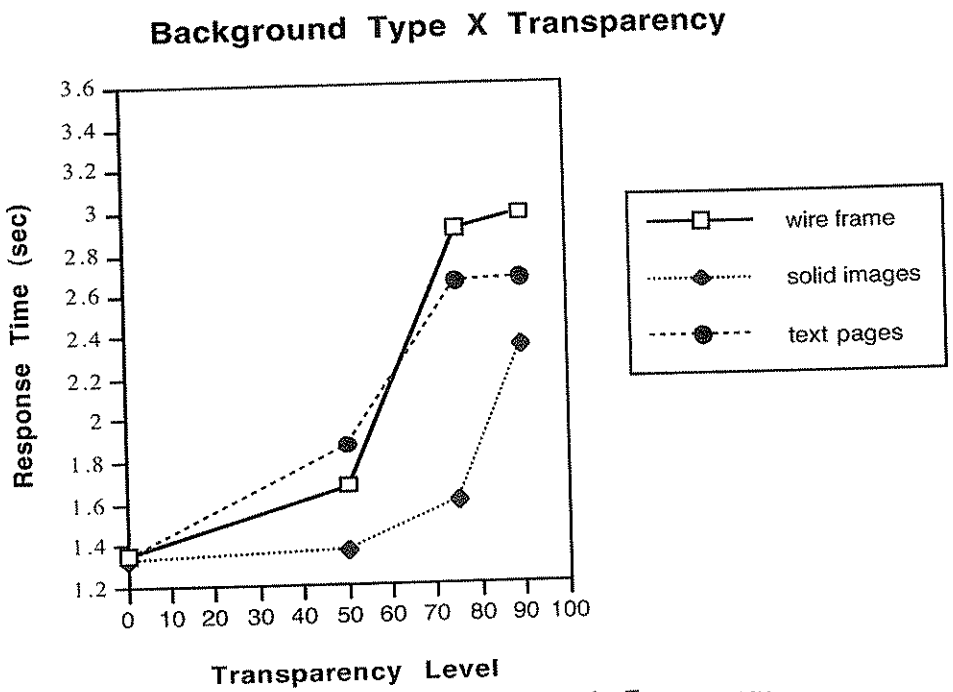


FIGURE 43. Mean Response Times for Transparency Levels X Background Types (across all icons types).

Using these results and the results presented in Figure 45, the Icon x Background x Transparency interaction reflects several interesting findings. When opaque palettes are used there is no interaction effect and no differences in performance between icon types or background types. Performance differences begin to emerge at 50% transparency, when a sensitivity to background content begins to have an impact. Wire frame and text page background perform significantly worse than solid imaages. At this level, performance is still insensitive to icon type differences. However, at the 75% transparency level and abov,e performance is impacted by both background type and icon type and the combination of the two. From 75% and more, solid images and solid icons are least affected by visual interference (performance is significantly better). Interactions involving either a solid background or a solid icon item are somewhat more affected but still perform significantly better than any combination which inclues wire frame stimuli or text stimuli on either the palette or background. Finally

For *overall* response time performance per transparency level (main effect – transparency) (Figure 44a), the groupings were: 90% + 75% (slowest), 50%, and 0% (opaque - fastest). (This provides a gross measure of the impact of transparency).

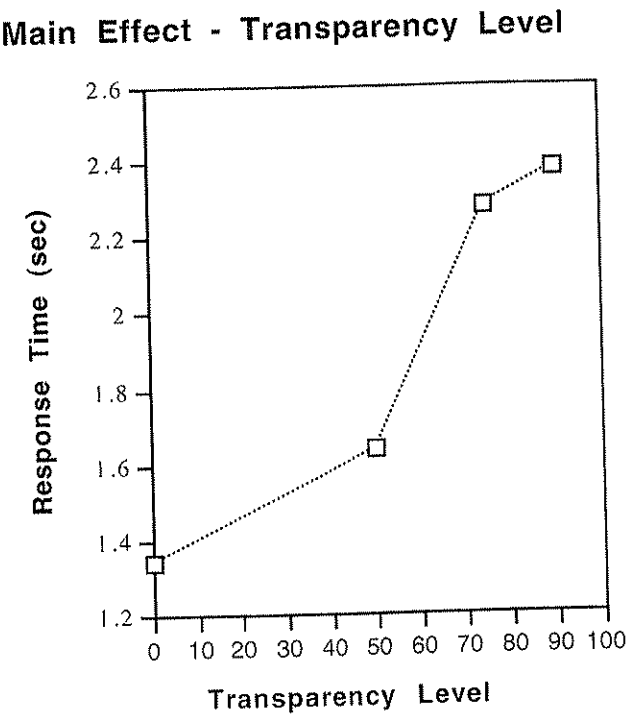


FIGURE 44 (a). Response Times for Transparency Levels across all Background Types and Icon Types. Note: there is no statistically significant difference between the 75% and 90% points.

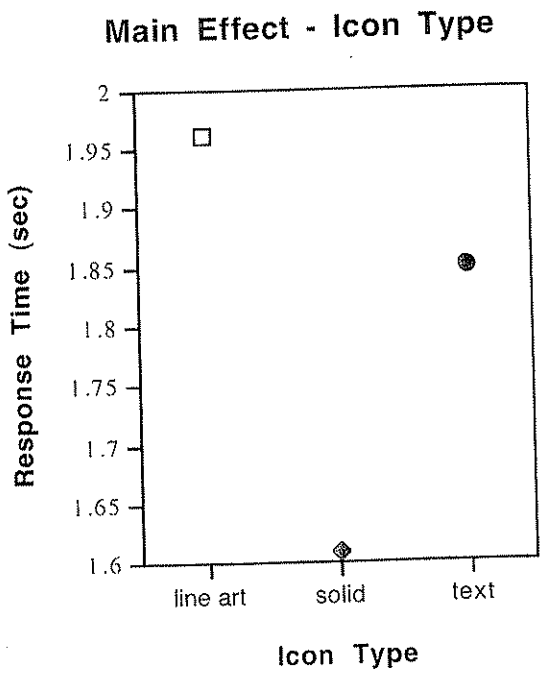


FIGURE 44 (b). Response Times - Icon Type across all transparency levels and background types.

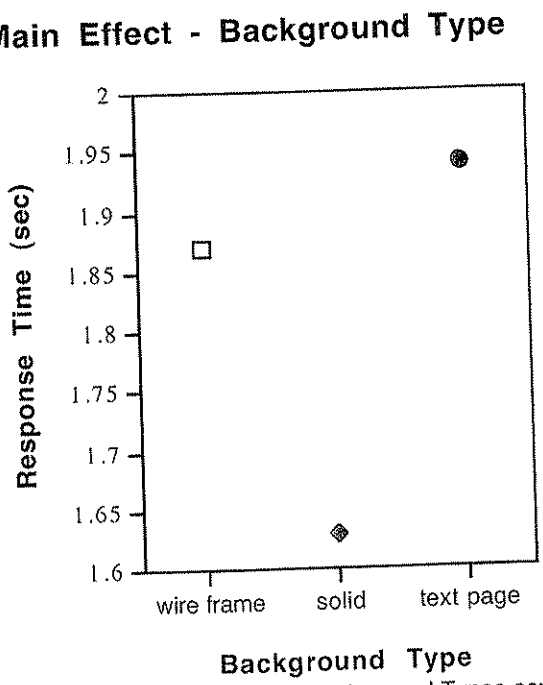


FIGURE 44 (c). Response Times - Background Types across all transparency levels and icon types.

	Background Groupings	Icon Groupings
fastest (good)	0% (opaque)	
	no difference	no difference
	50% Transparent	
	solid images	no difference
	wire-frame images text pages	
	75% Transparent	
	solid images	solid icons
	wire-frame images text pages	line art icons text icons
slowest (poor)	90% Transparent	
	no difference	solid icons
		line art icons text icons

FIGURE 45. Statistically Significant Groupings Within Transparency Levels

A detailed SNK analysis was run on the Transparency main effect (Figure 45, below). Note that from the graphs, we would anticipate that solids were grouped together and fastest, while line art and text performed similarly until highly transparent levels (75% and 90%). The graphs indicate that text icons performs better than line art. Figure 45 summarizes which points on the graphs are *not* statistically different (i.e., the clustering of data). It supports our basic assumptions, though wire frame backgrounds and line art performed more poorly than expected. This provides a way of confirming our earlier interaction analyses, shown in Figures 42 and 43.

For the Background and Icon main effects (across transparency levels) one would anticipate that 3 groupings would occur which represent the 3 types of items respectively (text, line art/wire frame, and solids). The statistically significant groupings of the SNK analysis of each Icon type and Background type are shown in Figure 46 (collapsed across all transparency levels).

	Background Groupings	Icon Groupings
fastest (good)	Camcorder solid image	All solid icons
	All other solid images Wire-frame human head Wire-frame camcorder	All line art icons All text icons
	All other wire-frame images	
slowest (poor)	All text pages	

FIGURE 46. Statistically Significant Groupings Across Transparency Levels

Targeting Error Results

Error trials were removed from the analysis of response time data and were subsequently analyzed separately. In total, less than 2% of the trials resulted in *targeting errors* or misses. This suggests that subjects were not guessing when targets were difficult to see. The breakdown of misses is shown in Table 2 and Figure 47 below.

In every case, targeting errors were due to substituting another icon of the same category for the target icon (e.g., an alternate incorrect text icon was selected instead of the target text icon). Text icon substitutions accounted for 30.1% of total targeting errors respectively, solid object icon substitution 29.7%, and line art icon substitution 39.7%. No targeting errors were due to accidental selection of adjacent items in the palette, suggesting the icon size used was adequate for selection accuracy. More targeting errors occurred at the 75% level versus the 90% level since users were more inclined to try selections at 75%. At the 90% level users typically marked

trials as illegible instead of attempting selection. Almost 80% of the targeting errors occurred at or above 75% transparency.

transparency level	number of trials	% of all misses
0% - opaque	5	4%
50%	20	16%
75%	53	44%
90% - mostly clear	42	35%

TABLE 2. Errors due to target misses

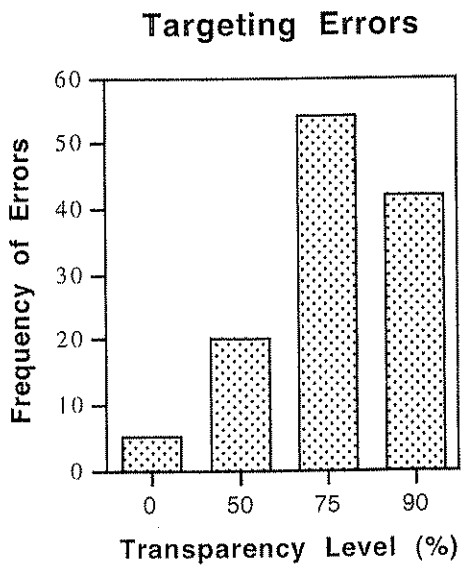


FIGURE 47. Graphical Overview of Targeting Errors

Legibility "Error" Results

The most frequent sources of "error" were trials that the subjects marked as "can't see" (which we believe prevented subjects from guessing randomly). In total, 18.4% of the trials were marked "can't see". The breakdown by transparency level is shown in Table 3 and Figure 48. Note that 55% of the legibility errors occurred at the 90% level and 99% occur at and above 75% transparency.

transparency level	number of "can't see" trials	% of total legibility errors
0% - opaque	0	0%
50%	20	1.3%
75%	350	23.6%
90% - mostly clear	1113	74.9%

TABLE 3. Trials marked as illegible

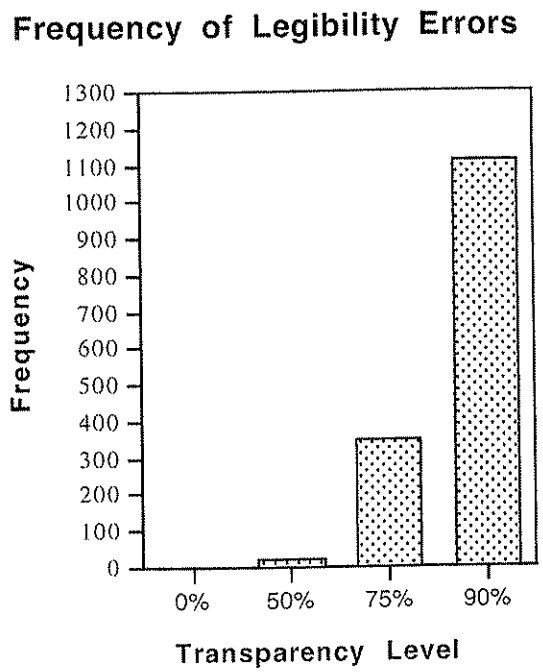


FIGURE 48. Graphical Overview of Legibility Errors

At the 90% level, *all* of the icon types appearing over *text or wire-frame backgrounds* were illegible. This accounted for 98% of the legibility errors at this transparency level (interestingly, only 2% of the illegible trials at 90% transparency were *solid* backgrounds). Further investigation showed that line art icon types appear the most problematic across transparency levels (42% of total). (Solid icons were 25% and text icons were 33% of total legibility errors respectively.) Finally, we determined the "threshold of frustration", the point at which subjects give up because the effort seems excessive. At 90% transparency, subjects marked trials illegible after attempting them for an average of 2.6 seconds, roughly twice the average trial time.

Qualitative Results

In general, subjects felt that they adopted a hierarchical "categorization" strategy for locating target icons. The categories were based on icon type (text, line art, solid) and then distinctive features within that type (e.g., shape, color, length of text). Subjects first searched for items belonging to the target icon category. They then used the most distinctive features to either locate the target icon or to eliminate the contender icons. All subjects felt the "line art" icons were the most difficult to discriminate since subjects had to search for tiny differences. Subjects commented that solid object icons seemed easier to find, independent of transparency level or background types. They used object color as a major cue. All subjects found the light bulb icon the easiest, primarily because no others had similar colors. Text icons were discriminated based on the shape of the words.

Wire frame backgrounds were perceived as most difficult for any icon type. The more dense wire frames were perceived as slightly easier. Subjects found the darkest solids were easiest (e.g., the camcorder) and commented that the palette seemed to "stand out" best on these images. Most notably, several subjects commented that after a number of trials the opaque palettes seemed "too bright" or "annoying" and that they were "used to the partial transparency".

Although the position of the target icon was randomly selected to avoid learning effects, subjects commented that they eventually learned the features of the entire set of 12 icons (without positional information). This reduced the time required to "study" the target when it was presented.

5.1.5 Discussion

As expected, icon type, background type, and transparency all affected response time performance (Hypothesis 1). Given the response time and error rates combined, 90% (highly transparent) palettes seem unusable (consistent with the Stroop Experiment results). In most cases transparency levels of 50% and 0% (opaque) seem to work about equally well, independent of icon type or background. This suggests minimal or no performance penalty for changing from an opaque palette to a semi-transparent palette. Our data support both our predicted performance curve (Figure 36b) (Hypothesis 1) and the proposed cut off points of 50% and 90% (Hypothesis 3). The curve and cut-off points were derived from the Stroop experiments. Therefore, our results here are consistent with those suggested by the earlier Stroop work.

Contrary to our expectation, wire frame backgrounds seemed to perform slightly worse than text (solid backgrounds were the best) (Hypothesis 2). The performance difference was more

pronounced with increases in transparency. Subjects commented that the wire frame images seem very visually complex and hence interfered the most. This also held for line art icons versus text icons: line art were as bad or worse than text icons as transparency increased. Our hypotheses overestimated the performance degradation resulting from text objects (relative to wire frame or line art). Both solid backgrounds and solid object icons are most resistant to interference and provide the best selection performance (consistent with Hypothesis 2).

It seems that the density of some of the wire frame background images skew (and improve) performance more towards solid image performance (e.g., the camcorder wire frame). Additionally, within the solid images, we believe that contrasting luminance levels improved performance on the mostly black camcorder background (the palette icons were colored or grays primarily). (This is consistent with a similar effect observed for color in the earlier Stroop Experiment.) This difference in luminance as it related to visibility was noted by several subjects.

All subjects commented about learning effects and categorization schemes used to facilitate locating items on the palette. Subject performance improved slightly over time and as familiarity with the set of icons improved. In a normal work context, we could assume that the most frequently used icons would likewise become well-known. Subjects would eventually learn which features distinguish the icons that they most frequently use. When this information is combined with consistent palette position, subjects will likely perform well, even if the icons are not clearly visible (i.e., highly transparent). A deliberate experimental design decision which generates more conservative measures is the choice to use randomized palette locations. While this does not reflect traditional palette design (context sensitive palettes aside) it does control for the large confounding effect of learning. Clearly, when items are presented in a predictable location, spatial memory will greatly facilitate target selection, independent of the visibility of the target itself. The borders of items seem to be a sufficient cue for expert users who have location mappings memorized. This is discussed in more detail in our later case study (Chapter 7).

One aspect of most experimental studies which may exaggerate the error rates is the cost or consequences of errors. In this experiment, although errors were logged, there was no "cost" associated with an incorrect guess. This may have led subjects to select items more often by guessing than one would observe in a real work environment where errors have consequences. Clearly, in any situations where there are attentional trade-offs which result in reduced performance, the designer must take into account the cost of missed events versus the cost of errors.

Finally, for simplicity and more experimental control we used static images to test our current attentional model. In more realistic applications, users would be moving the palettes and windows, particularly if the UI tools resemble ToolGlasses or MagicLenses. We know from preliminary prototyping (and the literature on visual perception) that motion parallax greatly helps users discriminate which features belong to which objects. We believe that the addition of motion will also benefit our transparent UI tools. This remains to be experimentally evaluated.

5.1.6 Summary

This experiment showed the performance effect of varying transparency in a icon tool palette. In general, there is minimal or no performance penalty for moving from opaque palettes to 50% transparent palettes, regardless of palette content or application domain (i.e., background content). Text items and line art items perform equivalently and are substantially more prone to visual interference than solid image objects (such as pictures, drawings, or perhaps video). With the addition of consistent positional cues and motion parallax (through palette movement), we anticipate that even highly transparent palettes will be usable, though this was not expressly tested in this experiment.

5.2 Experiment V- Text Menu Selection Task[2]

This section describes our experimental evaluation using variably-transparent, *linear text menus* superimposed over different background content: text pages, wire-frame images, and solid images (e.g., Figures 50a, 50b). As in the Icon Palette Experiment, this experiment also represents an extension to the Stroop studies. However, several important differences between this experiment and the Stroop experiments that should be noted. First, in the Stroop paradigm, the word naming task was a background focused attention task. In this case, the word identification task, i.e., the menu legibility task, is a foreground focused attention task. Second, instead of verbally naming the word, the task was modified to be an item selection task from the menu, using a mouse click to indicate the subjects' choices. This identification and selection task is more representative of traditional HCI applications. Third, both the items in the menu and the background images were visually complex. The menu items were complex by virtue of designing very similar text labels. The menu contains text items presented in either regular Motif-style fonts or a newly designed "Anti-Interference" (AI) font (Figure 50). We created AI

2 A summary of this experiment was published as Harrison, B. L. and Vicente, K. J. (1996). An Experimental Evaluation of Transparent Menu Usage. In the *Proceedings of CHI'96*, April 14-16, 1996, Vancouver, B.C., 391-398.

fonts by applying a contrasting luminance level outline around each letter, based on the ink color of that letter[3]. (This is described in detail in Chapter 6) Finally, the menus were presented at varying transparency levels, but the text was always visible. . This seems to be a more practical and reasonable design solution. Under the above described conditions, we evaluated the effect of varying transparency levels (from opaque menus to highly-transparent menus), the visual interference produced by different types of background content, and the performance of AI fonts.

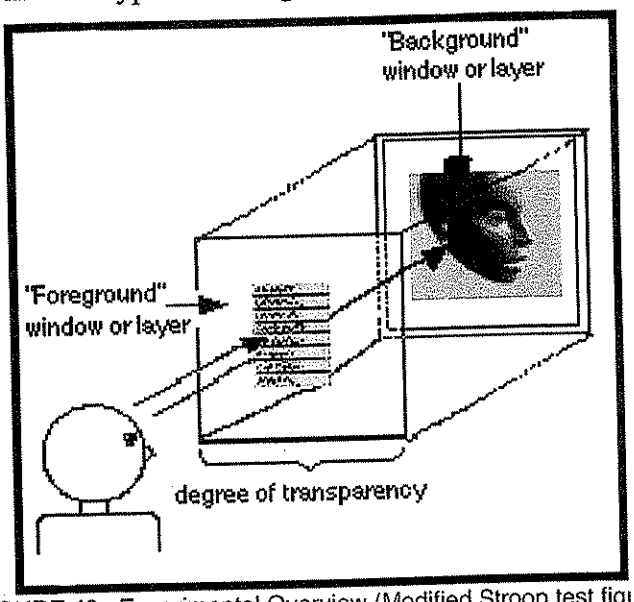
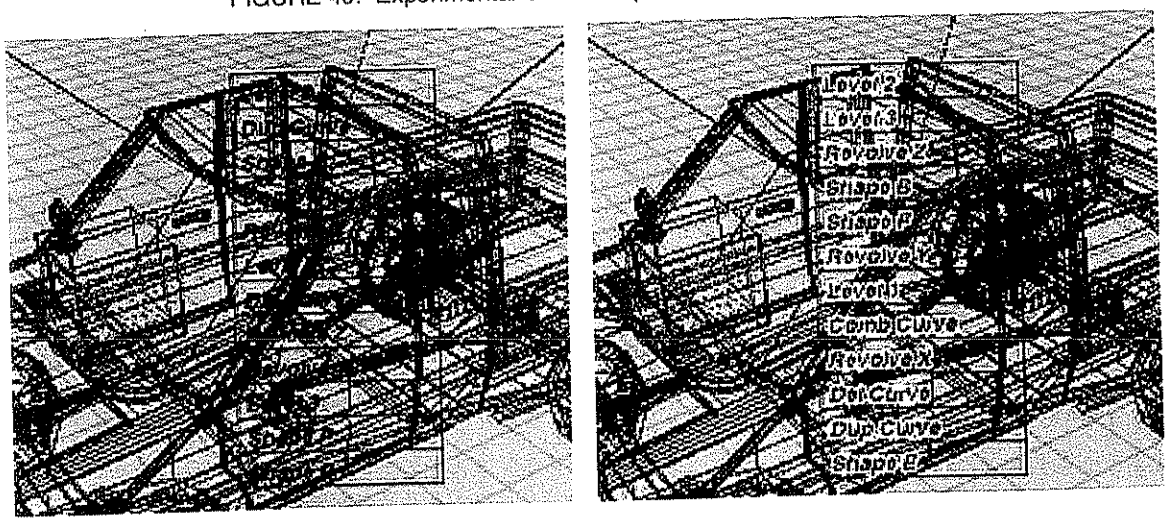


FIGURE 49. Experimental Overview (Modified Stroop test figure)

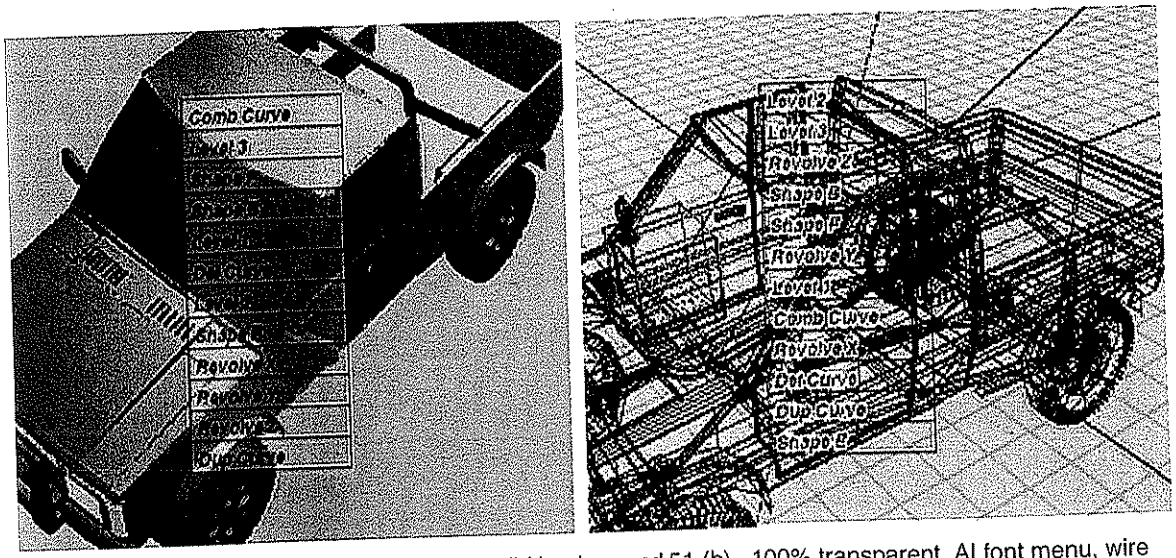


50 (a) Regular fonts, 100% transparent, wire background 50 (b) AI fonts, 100% transparent, wire background

FIGURE 50. Comparison of Regular and AI Fonts

3 This AI font development work was done in collaboration with Shumin Zhai, under a service contract to Alias Research Inc., patent pending.

The Icon Palette Experiment confirmed the viability of our methodology and provided a means for linking the theoretical Stroop results to a more realistic application–based experiment. However, icon palettes represent only one small subset of user interface tools. The study described in this section represents an "applied" experiment to evaluate transparent *menu* selection tasks, and can be applied to text legibility overall. One of the most general user interface mechanisms involves item selection from either a window or from a linear menu, where text labels are predominant. These text labels are either selectable in themselves (e.g., as menu items) or they are important cues in differentiating and identifying graphical window items for subsequent selection (e.g., radio buttons or sliders or field identifiers). In addition to windows and menus, we wish to eventually apply transparency to help systems and on-line documentation. Clearly the effect of transparency on text legibility is therefore a significant factor in the usability of such interfaces and tools.



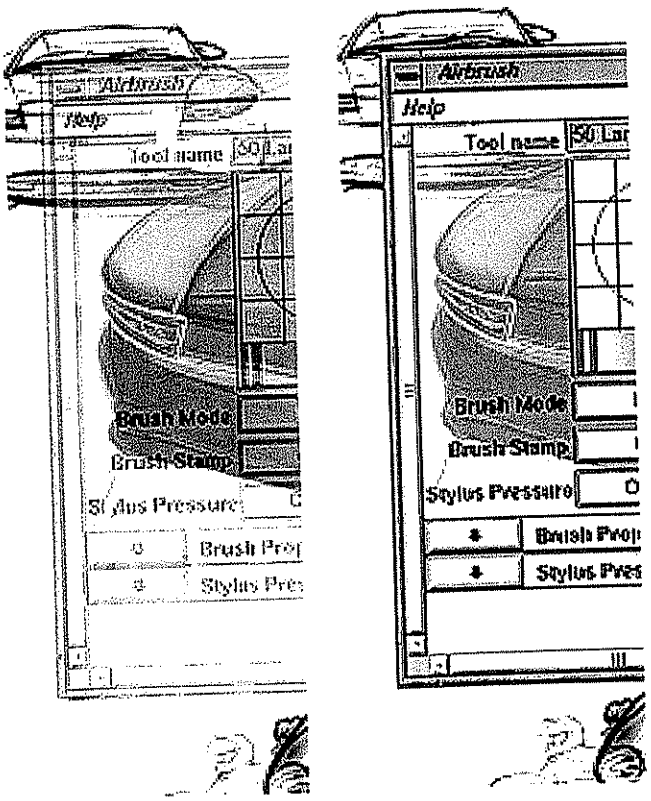
51 (a). 50% transparent, regular Motif menu, solid background 51 (b). 100% transparent, AI font menu, wire frame background

FIGURE 51. Experimental Sample Images

5.2.1 Predictions Based On Previous Experiments

The Text Menu Selection Experiment represents a focused *foreground* attention task and, as such, we anticipate a performance curve which resembles those found in the Icon Palette Experiment (i.e., opaque levels should have "good/fast performance" and transparency increases should degrade performance) (Figure 36b). This basically implies that the menu items will be sensitive to visual interference from the background as transparency increases. However, unlike the palette selection task where the *entire* icon was made transparent, the menu label remains opaque – only the surface area *around* the label is made transparent (e.g., Figures 51a, 51b). We believe that this difference reflects a realistic and reasonable design choice. In the palette

selection experiment, our icons were solid objects (as many icons typically are). In order to achieve a transparency effect, the icon image itself must be made transparent. In the case of text labels however, the text occupies only a small percentage of the "selectable region". Therefore, we may leave the text opaque and still achieve reasonable transparency using the remainder of the selectable area around it (e.g., Figure 52b). (Both design alternatives are shown in Figure 52. We feel Figure 52b represents the more realistic design choice. This was the method used in our experiment.)



51 (a). labels & surface around labels are both transparent (e.g.,the title bar, buttons on right hand side)

51 (b). labels are opaque, surface around labels is transparent (e.g., note the title bar, brush radius, & buttons)

FIGURE 51. Comparison of design alternatives for transparent text items. (Note that images are much better quality on a monitor.)

The menu selection task itself is a legibility or word naming task suggesting cut-off points similar to those from the Stroop Word Naming Experiment (Figure 38b). Best performance is expected to be maintained from 0% (opaque) to 50%; no further improvements occur beyond increases of 50% transparent (i.e., interference is no longer an issue). Poor performance is reached at about 90% (i.e., 10% of the foreground shows, 90% of the background shows) and does not deteriorate further. In this latter case, (as in the Stroop Experiment), interference strongly effects both response time and error rate such that the transparency levels between 90%

and 100% are virtually unusable. However, these cut-off points are based on experiments where the *entire* target object was transparent. We believe that the cut off points might shift right (towards higher transparency) on the hypothesized curve, given the opaque text labels – implying more resistance to visual interference. Based on extrapolating these previous experimental results, our predicted curves and cut-off points are shown in Figure 53.

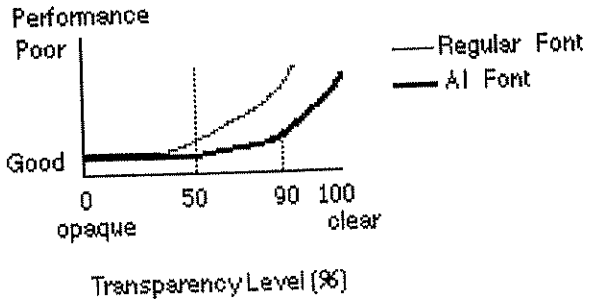


FIGURE 53. Projected curves - Menu Selection Experiment

When applying the Anti-Interference (AI) fonts, we anticipate more interference–resistant images. Therefore the low-end and high-end performance cut-off points would be higher (as depicted in Figure 53, "AI font" curve). (This is not unlike the effect shown in the Icon Palette experiment for analysis by Icon type and Background type, Figures 42 and 43 respectively, when the complexity of the image was simplified, for example, from text to solid).

5.2.2 Hypotheses

H1: As transparency level increases (i.e., the background is more visible through the menu) the response time and errors will increase (performance will be degraded due to higher interference).

H2: There will be a significant interaction effect between the background image type (text, wire frame, solid) and performance (i.e., legibility of the items).

We anticipate that increased complexity or information density on the background will make menu legibility decrease. Text backgrounds will have the worst performance, followed by wire-frame, then solid images.

H3: We anticipate that AI fonts will significantly improve performance by creating more interference resistant text.

H4: Lastly, we expect to observe proposed cut off points at 50% and 90%, reflected in the data.

5.2.3 Method

5.2.3.1 Subjects

A total of 10 students from the University of Toronto acted as subjects. They were pre-screened for color vision deficiencies and for familiarity with the product from which the images and items were taken. Subjects were paid for their participation and could voluntarily withdraw without penalty at any time.

5.2.3.2 Experimental System Configuration

The experiments were conducted on an SGI Indy™ using a 20 inch color monitor. Subjects sat at a fixed distance of 60 cm from the screen (average distance when working normally).

5.2.3.3 Experimental Design

A 2 x 9 x 5 fully randomized, within subject, repeated measures design was used. There were three independent variables: type of font, type of background, and transparency level. The types of font were: Regular menu and AI font menu. The types of backgrounds were: solid images, wire frame models, and text pages, with 3 unique samples of each (9 combinations). Transparency levels were based on our previous experimental results (see Chapter 4 and Chapter 5, Section 5.1) and test pilot results with this experiment. Levels of 0% (opaque traditional menus), 50%, 75%, 90% and 100% (clear) were used. The opaque level represented the baseline condition where the fastest performance was anticipated. In this case, recall that "clear" means the "surface" area of the menu is completely transparent while the text remains superimposed over this and hence is still visible. We used 6 possible target menu items (i.e., 3 pairs of matched items), without replacement, distributed within a 12 item menu. A total of 540 trials were run for each subject.

5.2.3.4 Stimuli

We used 2 groups of menu items: each group was visually similar to encourage subjects to read the entire item, rather than rely on diagnostic perceptual cues which would rapidly allow them to discriminate the location of a particular target item. The menu items were: Revolve X, Revolve Y, Revolve Z, and Dup Curve, Comb Curve, Del Curve. Six other menu items were distributed with the target items. (A 12 item menu was felt to be representative of the average menu size used within the target application domain; this was the median menu size with a range of 8 to 24 items). Items were randomly assigned positions within the menu for each trial. While this is

clearly an unrepresentative choice, it is more consistent with the focus of our experiment, which was text legibility. Having a fixed menu order would be more representative but it would allow participants to learn the position of different menu items, and thereby perform the task based on memory, not legibility. Randomizing the order of items on the menu thereby provided an unconfounded estimate of legibility performance. The target item was presented to the subject throughout the trial as a reminder. This was to prevent memory errors (which we were not testing for).

Using these 12 item menus, we assigned either regular font style or our AI font style to the menu text. Regular fonts were matched to the Motif style menus that appeared from windows on the SGI (Helvetica, 14 point, bold, italic was the best match). Height of menu items (i.e., the selectable rectangular region) was also matched to Motif standards. Pilot testing revealed that this size was easily selectable.

We developed Anti–Interference fonts as a potential interference resistant font technique (Figure 50b). Since an AI font has two opposing color components, it remains visible in any color background. The algorithm for AI fonts is described in detail in the Chapter 6 (the AI fonts and prototypes were developed concurrently with the experimentation).

We assigned background images of three types: text pages, wire frame images, and solid images. Three samples of each type were created for a total of 9 different backgrounds. Images were 8-bit color rendered images. These backgrounds were aligned such that a major portion of the content was directly under the menu, thereby ensuring the possibility of visual interference. The backgrounds were the same set used in the Icon Palette experiment (shown earlier in the Icon Palette experiment, Figures 38, 39, 40).

Levels of transparency were assigned to the menu based on our previous results and test pilot results with this experiment. Five levels were chosen: 0% (traditional opaque menus), 50%, 75%, 90%, and 100% (clear). The opaque level represented the baseline condition where the fastest performance was anticipated. To produce the exact image combinations required, we used alpha blending techniques as described in Chapter 4 (the same technique was used to produce the Stroop Experiment stimuli and the Icon Palette stimuli). This ensures precise combinations for any specified level, without graphical artifacts. For a more detailed discussion and comparison of the possible methods to produce transparency effects refer to Appendix A.

5.2.3.5 Procedure

The subjects' task was to locate and click on a particular target item in the text menu. The target menu item was presented to subjects throughout the trial at the bottom left of the screen (lower left corner, Figure 54), so that they would not have to rely on their memory. Again, this is consistent with the objective of obtaining an unconfounded measure of legibility performance. To prevent guessing, if the item on the menu was not legible, subjects were told to press a "can't see" button located at the bottom right of the screen (lower right corner, Figure 54).

FIGURE 54. Sample Trial Screen showing target item, stimulus image and "can't see" option

Subjects were given 20 practice trials. These trials were randomly selected with replacement from the set of 540 possible combinations. Subjects could ask questions if they wished during, or immediately after, the practice trials. For each trial, subjects were shown a randomly selected target icon and a button to start each trial ("Next Trial" button) which gave subjects time to study the target menu item prior to commencing each trial. The timer started as soon as this button was pressed, and the button press brought up a randomly ordered menu with a randomly ordered transparency level, superimposed over a randomly chosen background. Subjects had to locate and click on the target item within the menu. The timer stopped as soon as a menu item or the "Can't See" button was selected. Subjects could take short rest breaks whenever necessary and were debriefed at the end of the experiment.

Trials were presented in random order. The target menu item was randomly selected from the group, with replacement. Items were randomly assigned a position in the menu for every trial. A font type (regular or AI font) was randomly assigned, with replacement, for the font style of the menu. We randomly assigned background image types and a sample of the type. Finally we randomly assigned the level of transparency to the menu. All 6 x 2 x 9 x 5 possible combinations were used (targets x menu type x background x transparency). A total of 540 trials were conducted for each subject. Each session lasted about 45 minutes (one session per subject).

Selection response times and errors were logged. Response selections were made using the mouse by clicking on either a palette item or by clicking on the "Can't See" button. Two error conditions were possible: the subject pressed the "can't see" button indicating that the item was not legible, or the subject selected the incorrect menu item. In the latter case, the item selected and its location were logged. Error trials were removed from subsequent analysis of response time data and were analyzed separately. At the end of the experiment, subjects were asked which items they felt to be "easiest" and "hardest" and if they were aware of employing any particular strategy. The experimenter additionally noted any observable strategies based on movement patterns. Subjects' responses, any open ended comments, and experimenter observations were recorded and are reported below.

5.2.4 Results

We have categorized our results by Response Time analysis, Error analysis, and comments from the interviews with subjects. A 2 x 3 x 5 univariate ANOVA was carried out on the selection response time data with Font type, Background type, and Transparency as the independent variables. Background types were also analyzed by each individual image (9 in total) in a subsequent 2 x 9 x 5 ANOVA, reported below.

Quantitative Statistical Analysis - Response Time

Highly significant main effects were found for all of our major variables: Background, Transparency, and Font (Table 4), as proposed in Hypothesis 1. Main effects are shown below in Figures59 a, b, and c. However, we are primarily interested in the transparency and font effects and their interactions with background type. Statistically significant interaction effects were found for Background X Font (Figure 56), Transparency x Font (Figure 57), and Background x Transparency (Figures 58a, b, c). Finally, a significant 3-way interaction between Background x Font x Transparency was found (Figures 55a and 55b). These results are summarized in Table 4.

We are primarily interested in the 3-way interaction, in the Transparency x Font effect and in the interactions with Background type (as opposed to the main effects).

condition	df	F value	p<
background type	8, 72	1.06	.01
transparency level	4, 36	4.12	.0001
font type	1, 9	3.38	.0001
background X font type	8,72	1.59	.001
transparency X font type	4,36	20.57	.0001
bkgrnd type X font type	8, 72	1.59	.001
bkgrnd type X transp	32, 288	2.44	.01
bkgrnd X font X transp	32, 287	3.76	.001

TABLE 4. Statistical Results for Main Effects and Interactions

To determine if the differences are significant between the individual lines plotted within each of the graphs, a Student-Newman-Keuls (SNK) test was run post-hoc as a comparison of means, with an alpha level = 0.05. (This determines the clustering of items within font type, background type, and transparency level and indicates which items are not statistically different.) The results of the post-hoc analyses are described below.

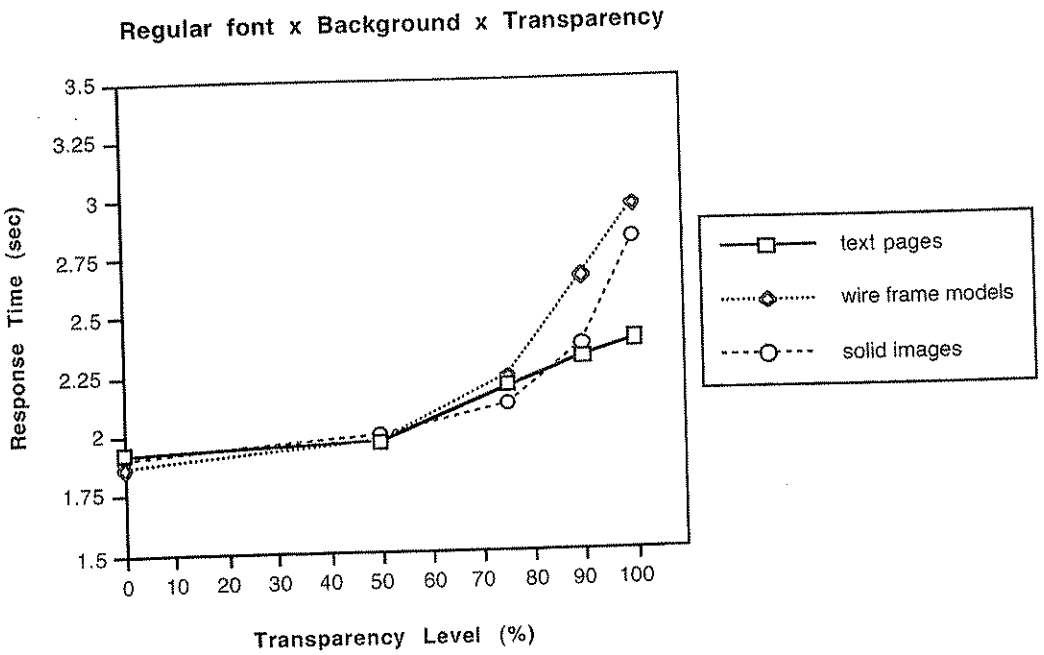


FIGURE 55 (a). Mean Response Times for 3-way interaction using Regular Font x background x Transparency (by background types)

Post-hoc SNK analysis of the Regular Font x Background x Transparency (Figure 55a) revealed that there were no statistically significant differences between the background types (Figure 58a), when collapsed across samples (i.e., 2 x 3 x 5 analysis). To further clarify the Background x

Font interaction and the Background x Font x Transparency interaction, we carried out a 2 x 9 x 5 ANOVA, modified to now include all of the individual samples of background images. We additionally normalized the data (since several conditions had resulted in high variance) and carried out a further 2 x 9 x 5 ANOVAand SNK post-hoc analysis. This permitted us to examine attributes which might relate to specific image properties of particular stimuli (which would have been averaged out in Figure 55a). Based on this analysis, we found that there were no statistically significant differences between any of the 9 backgrounds at the 0% and 50% level. At 75% transparency, the text page background which duplicated the menu font performed significantly slower than other 8 background conditions (i.e., background and foreground fonts were matched). The solid image which consisted of light–colored elements performed significantly faster than other 8 backgrounds. At the 90% transparency level, performance on the matched text page was significantly worse than all other backgrounds. However, the non-matched text pages were significantly faster than other background types from 90% transparency upwards (i.e., 90% and 100%). In general, the wire frame models were significantly worse than the solid images. At the 100% transparency level, notice a sharp increase in the solid background plotted in Figure 55a. This can be attributed to the significant difficulties caused by the almost black camcorder image (under the black fonts). From this more detailed analysis, we observe that the Regular fonts are significantly influenced by specific background image properties above the 50% level. At the 100% level, dark components on the background have a dramatic performance impact.

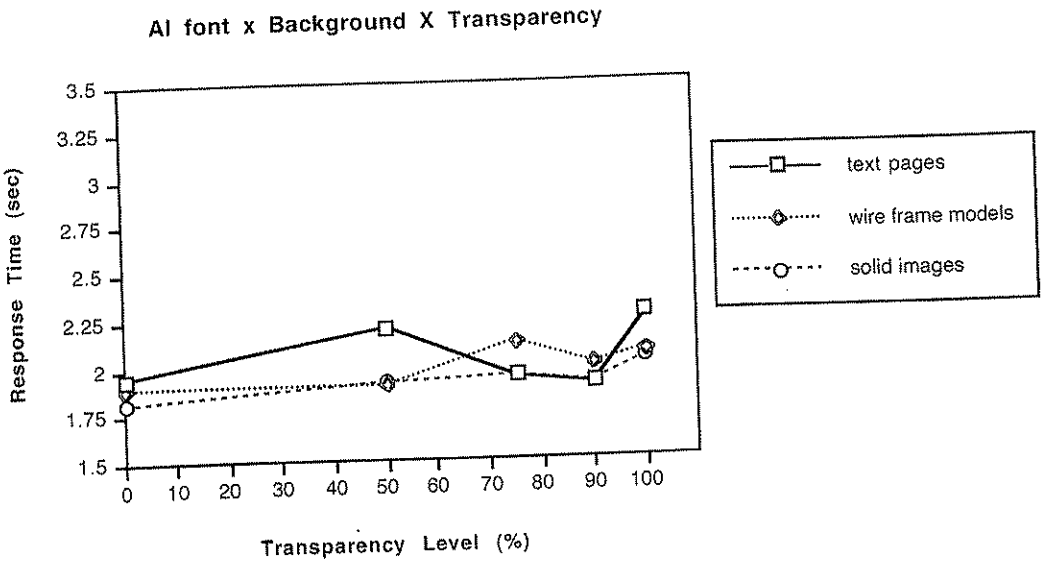


FIGURE 58 (b). Mean Response Times for 3-way interaction using Al Font x background x Transparency (by background types)

Post-hoc SNK analysis of the AI font x Background x Transparency interaction (Figure 55b) was carried out. Overall, AI fonts were generally less sensitive to the type of background (Figure 55b). Subsequent SNK analysis by Background type showed no significant differences for the AI fonts (Figure 55b), with the exception of text pages at 50% and at 100% transparency. We carried out further 2 x 9 x 5 ANOVA, a normalized 2 x 9 x 5 ANOVA, and SNK post-hoc analyses to determine more precisely how the specific background image properties affected performance. The performance variance for the text pages was substantially higher, even when the data were normalized, which we believe accounts for the difference at the 50% level . However, there were no significant differences in the variance at the 100% transparency level between any of the backgrounds. Variance alone cannot explain this significant performance difference. It would appear that at the 100% (clear) menu condition, when using AI fonts, text page backgrounds result in significantly more interference than either solid images or wire frame models and hence performance is degraded.

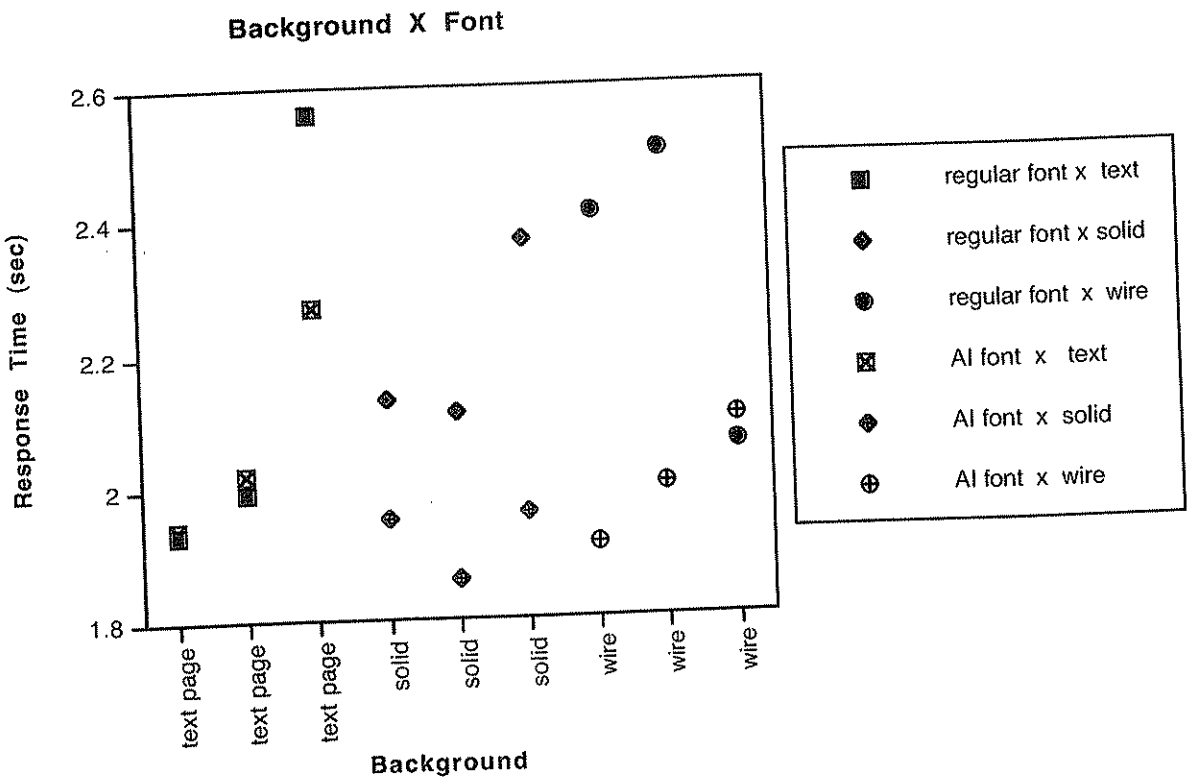


Figure 59. Mean Response Times for Background X Font interaction (collapsed across transparency level)

In examining the data for the Font X Background effect (Figure 56), significant interaction effects were found for the 2 x 9 x 5 ANOVA (i.e., individual background samples used). The solid symbols in Figure 56 represent the regular font, while the cross symbols are the AI font

results. The important thing to observe is that the general relationship between individual image samples within a given type and performance is consistent regardless of font style (with one exception in the last wire image). By this we mean that comparing within text samples, within wire frames, and within solids, the same sample performed best, worst, or in the middle in ranking - for either font style. We partially attribute the improved performance in the regular font X 3rd wire frame condition to the sparseness of that particular wire frame image. It is unclear why this did not have an equal impact on the AI font condition. However, the effects of transparency level is of more interest in this experiment and therefore subsequent analysis focused on the Transparency variable.

We ran a finer grained SNK analysis on font type *at each transparency level*. At the opaque menu level (0%) and the 50% transparent level, there were no statistical differences between font styles (AI versus regular). At 75%, 90%, and 100% transparency the AI font performed significantly faster than the regular font (as shown in Figure 57, below).

There were statistically significant differences for the regular font between the following transparency levels: 100% - poorest, 90%, 75%, and 50% + 0% (which performed equivalently well). (This finding is consistent with our earlier Stroop Experiment results and Icon Palette Experiment results.) There were statistically significant differences within AI fonts between the following transparency levels: 100% - poorest, 90% + 70% + 50% + 0% (not different). Note that AI fonts were relatively insensitive to the type of background. However, the matched text page showed the worst performance. Other background images were not significantly different.

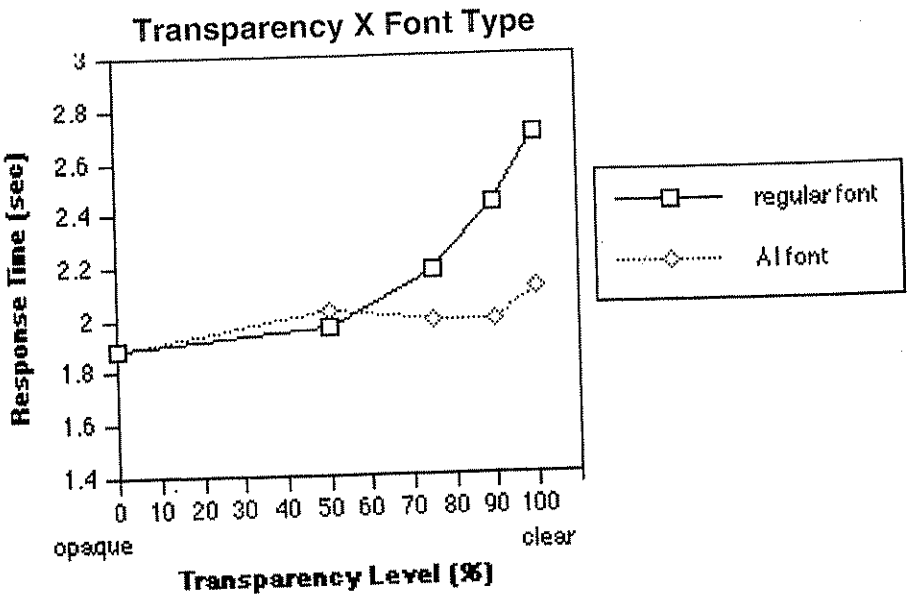


FIGURE 57 Mean Response Times for Transparency Levels X **Font Style** (across all background types)

A finer grained SNK analysis of background type data, by transparency level (Figures 58 a, b, c) showed no statistical differences between 0% and 50% for any background type. There were significant differences between background types at transparency levels > 50%, though these differences are not based on the category (text, wire, solid) but rather on the individual image properties. For example, the text pages each used a different font style, one of which was Helvetica 14 bold (purposely matching the menu item font style). This page performed significantly slower than the other pages (Figure 58a). Data varied more widely on the text background conditions, partly as a result of whether the target item appeared in coincidental alignment with a paragraph break (i.e., a blank spot in the page).

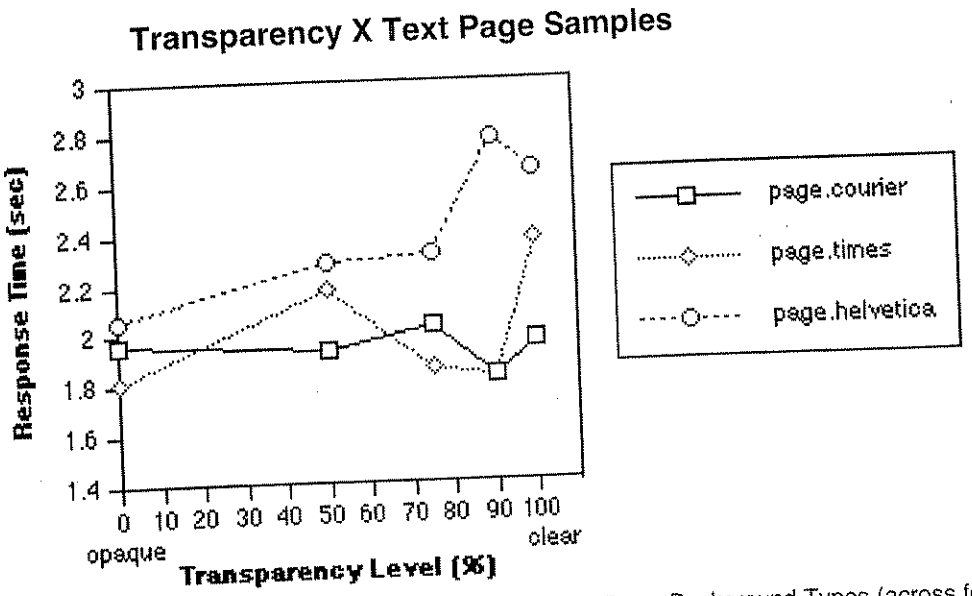


FIGURE 58. (a). Mean Response Times for Transparency X **Page** Background Types (across font types).

Analysis of wire frame backgrounds (Figure 58b), revealed that the denser wire frame images (i.e., more complex meshes and therefore also darker in color) performed significantly slower than the simpler wire frame.

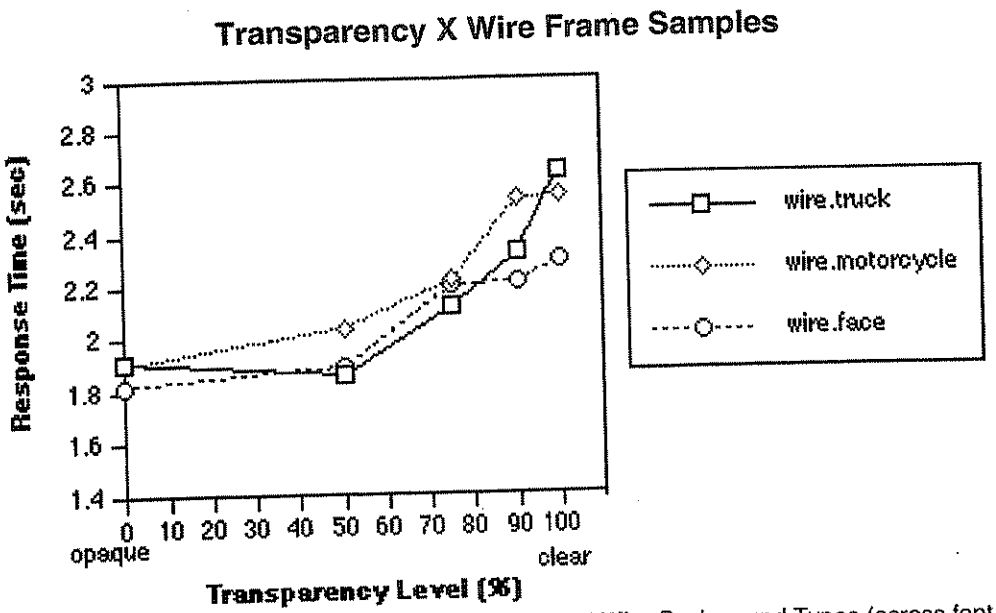


FIGURE 58. (b). Mean Response Times for Transparency X **Wire** Background Types (across font types).

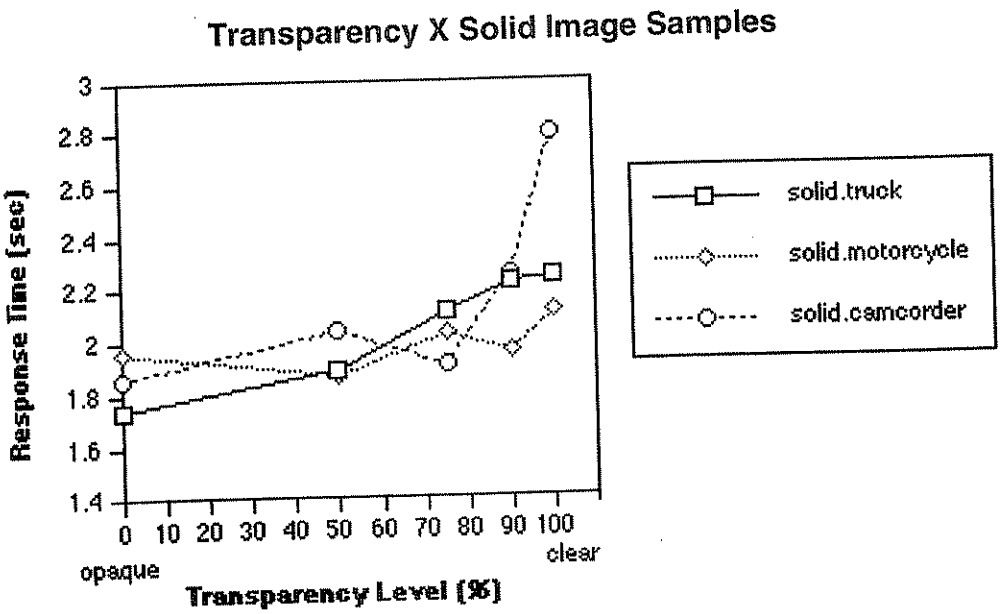


FIGURE 58 (c). Mean Response Times for Transparency Levels X **Solid** Background Types (across font types).

The solid images with black components (the truck and the camcorder), performed significantly slower than the solid multi-colored motorcycle image. (These images were used to create a worst case scenario of black menu item fonts over partially black background content.) At the 100% (clear) level, images with black components were clearly the most difficult (Figure 58c).

Finally, we observe the anticipated results when we examine the graphs for the main effects (Figure 59 a, b, c). Performance does indeed appear to deteriorate as transparency level

increases (Figure 59a). In general, the AI font is significantly faster than the regular font (Figure 59b). Lastly, solid images perform faster than either wire frame background or text pages, and somewhat surprisingly, wire frame background perform slowest (Figure 59c).

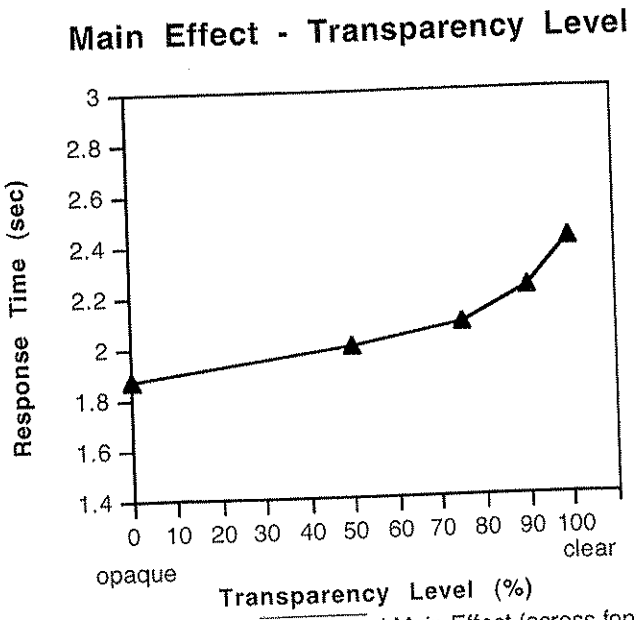


FIGURE 59 (a). Transparency Level Main Effect (across font type and background type)

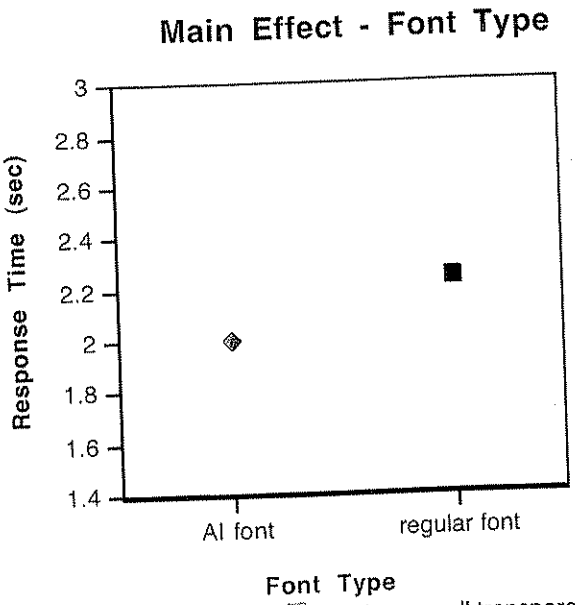


FIGURE 59 (b). Main Effect for Font Type (across all transparency levels and background types)

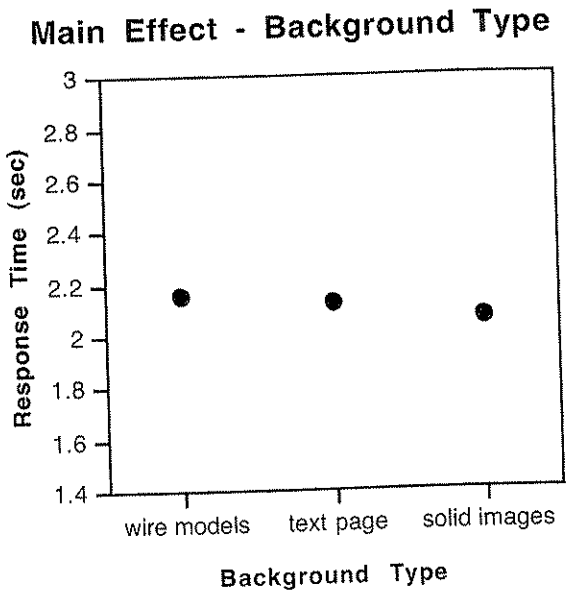


FIGURE 59 (c). Main Effect for Background Type (across all font types, transparency levels, and samples within each background type).

Targeting Error Results

Error trials were removed from the analysis of response time data and were subsequently analyzed separately. In total, 1% of the trials resulted in *targeting errors* or misses. There were two types of targeting error possible: accidental selection of an adjacent menu item (45% of total) and substitution of an incorrect menu item (55%). In the latter case, the subject incorrectly selected a menu item by replacing the target item with a similarly named item such as Revolve X instead of Revolve Y.

The adjacency item errors are most strongly influenced by the width of the target areas. This was matched to standard Motif menu widths and so we did not wish to increase the width size, reducing these errors. However, we were most interested in substitution misses. These can be partially attributed to poor visibility of the target item in the menu and mis-reading the menu. Several subjects immediately realized when they had made such substitution errors ("oh no, I meant to select Revolve X, not Y"). The breakdown of substitution errors is shown in Table 5, below. These errors were surprisingly evenly distributed across transparency levels. This implies that the errors may be due to fatigue rather than poor visibility.

transparency level	number of trials
0% - opaque	9
50%	5
75%	6
90%	3
100% - clear	8

TABLE 5. Errors due to target misses by substitution

The substitution errors were also evenly distributed across font type for each level. AI fonts made little difference in reducing these errors.

Legibility "Error" Results

Of the total error trials, 48% were those that subjects marked as "can't see" (which we believe prevented subjects from guessing randomly). In total, less than 1% of the overall trials conducted were marked "can't see". The breakdown by transparency level is shown in Table 6. Note that almost all of the legibility errors occurred at the 100% level. All of these illegible trials were in the regular font condition (i.e., no AI font trials were marked illegible).

At the 90% level, menu items appearing over the text page or wire frame backgrounds were illegible. Solid images showed no errors at this level. At the 100% level, the two solid backgrounds with black color components accounted for *70%* of the illegible trials. (The menu font was black and therefore one would expect these trials to be illegible due to the black on black imagery). However, surprisingly, text pages accounted for only 3% of the errors made at this level.

transparency level	number of "can't see" trials	% of legibility errors
0% - opaque	0	0%
50%	0	0%
75%	0	0%
90%	6	10%
100% - clear	51	90%

TABLE 6. Trials marked as illegible

The mean response time for legibility errors was 6.84 seconds (the "threshold of frustration" where subjects give up), almost 3 times the response time for other trials. This implies that subjects exerted substantial effort to respond to each trial before giving up. In effect, this figure represents the "tolerance threshold" beyond which it is too much effort to locate the target.

Qualitative Results

Subjects commented that the wire frame backgrounds seemed the most difficult in general and the solid backgrounds were the easiest. However, subjects also noted that highly transparent menus over black backgrounds were very hard. Most subjects commented that even a small change in the transparency level (from 100% clear to 90%) made a substantial difference in these black–on–black conditions. This change allowed subjects to see and select items where previously they marked the trial "can't see".

Subjective preference seemed to favor changing the transparency level to improve visibility, as opposed to changing to the AI font. Several subjects commented that they did not like the "outline font" and if given a choice, preferred the 50% transparency level. This seemed to be an aesthetic preference, unrelated to perceived performance.

5.2.5 Discussion

As reflected in Figure 59a, transparency levels significantly affected response time and error rates (independent of font type or background), supporting Hypothesis 1. Also, we found evidence to support our predictions on Regular font performance and AI font performance (Hypothesis 3). The AI fonts produced a substantially flatter performance curve, shifted towards better (i.e., faster) performance as transparency increased. This implies that AI fonts are more interference resistant. The real advantage of using AI fonts was only realized at higher transparency levels (i.e., over 50%). In fact, AI fonts at 75% and 90% transparency levels produce results similar to those of using unmodified fonts at 50% transparency. This result might be used in the design of text–based transparent interfaces.

Menu selection X background content interactions were most strongly affected by higher transparency levels. There are not large performance differences between 0% to 50% transparency (Hypothesis 4). (This is consistent with results from our Stroop Experiment and the Icon Palette Experiment.) Surprisingly, the text backgrounds produced much better performance than expected (contrary to Hypothesis 2 predictions), particularly in the case of mismatched background page fonts to foreground menu fonts. The most critical dimension of interference with text menu selection tasks is color. The closer in shade and hue the background is to the

menu text color, the higher the interference and the poorer the resulting performance. In addition to image properties, the text pages produced highly variable performance. This was partly due to the coincidental alignment of some targets with paragraph breaks or blank spaces within the page. Solid background images did not perform as well as expected, relative to other background types.

Finally, we did observe the hypothesized cut–off points at 50% and 90% (Hypothesis 4) for both the font type variable and for the background type variable. In each case, there is minimal or no performance penalty for moving from opaque menus to 50% transparent menus.

5.2.6 Summary

The Text Menu Selection experiment showed that Anti-Interference (AI) fonts produce interference resistant text which produces stable performance for levels of transparency from 0% (opaque) to about 90% transparent, independent of background content. Regular Motif style fonts work well for levels up to 50%, and are, in fact, preferred over AI fonts. However, at levels beyond 50% transparent, regular fonts show a substantial decrease in performance in terms of both selection response time and errors. Furthermore, contrary to expectation, individual image properties and, in particular, color properties, rather than image type (such as text, wire frame or solid), have a substantial performance impact.

Again, as in the Icon Palette experiment, we would anticipate that consistent positional cues (as opposed to randomized locations) would greatly compensate for loss of visibility in highly transparent conditions. Also, the ability to move the text menu or text items would provide a motion parallax cue, thereby helping to reduce the visual interference between the foreground and background content. Neither of these conditions were part of the scope of this experiment.

5.3 Experiment VI - Background Selection Task

The two previous experiments were designed for icon palette and text menu selection tasks (i.e., focused foreground attention tasks), while preserving an *awareness* of the background. The results are appropriate for and applicable to selection of foreground items. However, we might question how one could measure "awareness" of a background. Also, many tasks have requirements for more than just awareness of the background. For example, the ToolGlass system (Bier et al, 1993; Stone et al, 1993) requires *alignment* of the palette item with a specific background object or area. In this case, background visibility requirements are higher than that required for simple awareness. Clearly, the goals of the task must be taken into account in order

to assess the priority of foreground versus background visibility. To better understand these differences, studies on background selection or alignment tasks are also needed. These studies complement the foreground task selection experiments described earlier.

In the experiment described in this section, we evaluate the extent of background awareness or visibility again using fairly realistic work contexts. This compares roughly to the Word Naming Experiment of the Stroop experiments (also a background focused attention task) with several modifications. As in the Icon Palette and Text Menu experiments described previously, our goal was to introduce progressively more representative elements into our experiments. To this end, we used icons, text menus, and background images taken from real products.

In order to maintain consistency with comparable foreground attention tasks (reflected in the previous two experiments of this chapter), we used identical stimuli, the same basic methodology, and a similar target selection task, with the following modifications. Subjects now ignored the foreground and focused their attention on the background. Subjects did not have a "can't see" option in this experiment. We wanted to determine the extent of the contribution of contextual cues to the selection task. This was to be assessed from the opaque menu and opaque palette condition results, and from the interview data. Subjects were instructed to make an educated "best guess" in target selection in these cases. Finally, our baseline (best case performance) condition was modified to be a background only selection task (no foreground images were present). This roughly corresponds to the word only condition in the Stroop Word Naming experiment. In addition to response time data, we also logged the distance of the selection point from the target (i.e., a bulls-eye error ranking approach). This distance was used to determine hits or misses and to determine the magnitude of the errors, based on a threshold radius around the target. Other than these necessary alterations, our methodology remained consistent with prior experiments.

We believe that the feature selection task reflects many of the components which would constitute "awareness". Subjects maintain awareness by casually monitoring global states or state changes and using various contextual cues to help inform them about objects which are not totally visible. In our experiment, we apply similar contextual cues with the proviso that subjects are explicitly and deliberately focusing on a potentially poorly visible image (as opposed to being peripherally aware while focusing attention elsewhere). The subjects' ability to accurately select features does reflect the extent of visibility and hence the maximal awareness they are able to achieve.

5.3.1 Predictions Based on Previous Experiments

Based on the results of the Stroop background focused attention task (Word Naming experiment), we might expect results shown in Figure 60a. Performance is poor when the foreground layer is more opaque. As transparency levels increase, and hence background visibility improves, selection time improves and errors are reduced.

If we further modify our prediction based upon what we learned about image interaction effects from the Text Menu (Figure 60b) and the Icon Palette (Figure 60c) experiments, more precise cut-off points and interaction effects can be predicted. (These prior experiments were foreground tasks and as such, we expect the *inverse* curves to apply to background tasks.) This suggests the relative response time performance of the different background types (shown in Figure 61a).

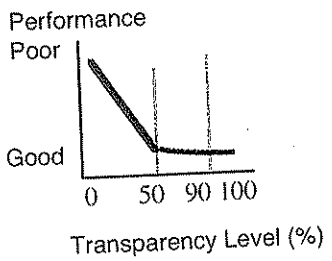


FIGURE 60 (a). Word Naming Experiment curve, background focused attention task

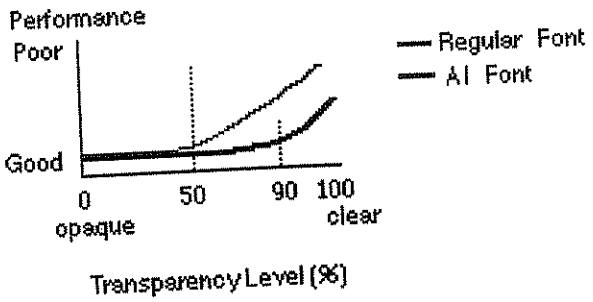


FIGURE 60(b). Menu Selection Experiment focused foregroundattention task

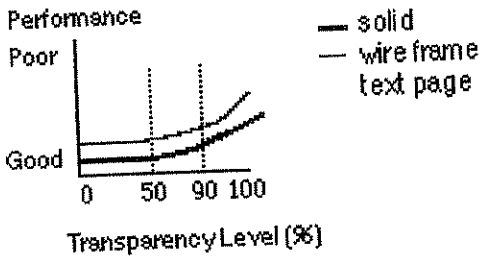
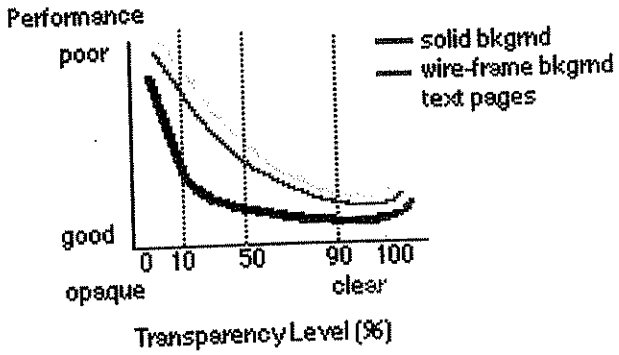
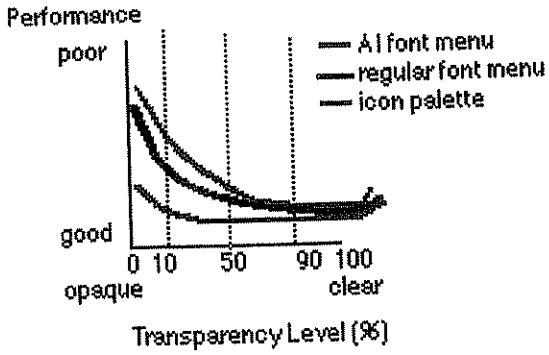


FIGURE 60(c). Icon Palette Experiment focused foreground attention task

We additionally predict that the information density of the foreground image will have an interaction effect. In particular, text is more complex than solid images and therefore the text menus of either type should perform worse than the icon palette. AI fonts are slightly larger than regular fonts, therefore the AI menus might be expected to cause more interference, resulting in poorer performance (Figure 61b).



61(a). Predicted results for background selection experiment through Icon Palette



61(b). Predicted results for background selection experiment through Text Menu

FIGURE 61. Predicted Performance

Finally, based on the design of *this* experiment, we would predict that the baseline (background only) condition might be slightly *more* difficult than a highly transparent menu or palette. This is because we guarantee to subjects that the target occurs under any menu or palette, if present. The boundaries of the foreground image may restrict the target search space whereas no such borders exist in the background only baseline condition. It is unclear how much advantage this will create. We expect that accuracy would continue to increase, with a maximum at 90% transparent (i.e., this boundary cue will effect accuracy), as reflected in Figure 61b at the 100% level.

5.3.2 Hypotheses

H1: As transparency level increases, performance will improve (i.e., the background is more visible through the foreground). Therefore, the response time and errors will decrease. At 100%, as mentioned previously, we anticipate slightly slower response times since the search space is not bounded.

H2: We anticipate that increased complexity or information density on the foreground will make target legibility decrease. Anti-Interference text menus will have the worst performance, followed by regular text menus, and lastly icon palettes.

H3 The content of the background image (text, wire-frame, solid images) will also have an effect on performance (response time or accuracy). We believe that solid images will have the best performance, followed by wire frame images, and lastly text pages.

H4: Lastly, we expect to observe our proposed cut off points at 10% and 50%, reflected in the data.

5.3.3 Method

5.3.3.1 Subjects

A total of 14 students from the University of Toronto acted as subjects. They were pre-screened for color vision deficiencies and for familiarity with the product from which the images and items were taken. Subjects were paid for their participation and could voluntarily withdraw without penalty at any time.

5.3.3.2 Experimental System Configuration

The experiments were run on an SGI Indy™ using a 20 inch color monitor. Subjects sat at a fixed distance of 60 cm from the screen (average distance when working normally).

5.3.3.3 Experimental Design

A 9 x 3 x 6 fully randomized, repeated measures, within subject design was used. There were 3 independent variables: type of background, (3 levels) type of foreground (3 levels), and transparency level (6 levels). The three background types were: text pages, wire frame images, and solid images. There were 3 different orientations of each of these (9 backgrounds in total). The 3 foreground types were: Motif font text menus, AI font text menus, and icon palettes. The 6 transparency levels were: 0%, 25%, 50%, 75%, 90%, and 100%. Three replications of the data set were used giving a total of 486 trials for each subject.

Dependent variables of selection response time (measured in seconds, based on a mouse click) and distance from target were logged. We classified hits as being within 20 pixels in any direction of the center of the target. This corresponds exactly to the size of the highlight box displayed in the target image. Pilot testing showed this to be a reasonable minimum size. Error trials (i.e., distance > 20 pixels) were analyzed separately.

Each session lasted about 45 minutes.

5.3.3.4 Stimuli

To ensure consistency, the same set of stimuli from the Icon Palette experiment and the Text Menu experiment were combined. There were the same 12 icons distributed on the palette, arranged in 3 rows by 4 columns. We re-used the same 12 item text menus Text menus were drawn in either regular or Anti-Interference (AI) font style. Randomization of the foreground images (i.e., menu or palette) was done to ensure no accidental confounding effects were introduced through inadvertent alignment of foreground objects to background targets.

One difference betweent this experiment and the two preceding experiments was that for each image, we generated three slightly different views or orientations so that the alignment of background to icon palette was randomized. This meant that the target location could not be predicted based solely on the image type. All backgrounds were aligned such that a major portion of the content, and in particular the target item, were aligned directly under the palette or menu. This resulted in 27 different background images.

Targets were selected in such a way that they were guaranteed to be under the palette or menu, should one be present. We selected targets which were reasonably realistic and identifiable features such as specific text words, a rear view mirror on the truck, the tip of the nose on the face. These would normally be the type of item a user would manipulate in the real product applications. Target stimuli were created by a separate program which displayed each of the possible backgrounds with the menu and palette borders superimposed simultaneously (i.e., empty menu and palette) at 100% transparency (clear) (see Figure 62a). The experimenter selected a variety of features which were aligned under both the palette and the menu. Image files were rotated, shifted, and scaled to facilitate this alignment. Target specification was done a single time to generate the targets for the entire experiment. All subjects used the same targets. Some targets were chosen to be well-defined objects with discernible object boundaries (such as a rear-view mirror). Other targets were selected which did not have clearly discriminable boundaries (no color, shading, or edge cues), such as the tip of the nose on the solid face image. In all cases, during the experiment, the targets were highlighted with a 20 pixel X 20 pixel box surrounding them. This box was a red and yellow border (Figure 62b). Two color borders were used to ensure that the highlighting stood out, regardless of the color components of the image file. Context was provided for the target feature by showing a portion of the total image which surrounded the target (the target always appeared centered).

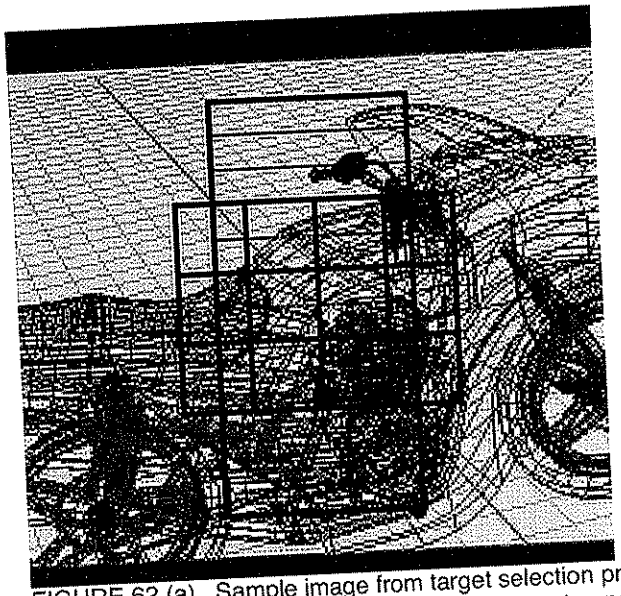


FIGURE 62 (a). Sample image from target selection program. Empty Menu and Icon Palette borders are shown to ensure target is aligned under them when displayed.

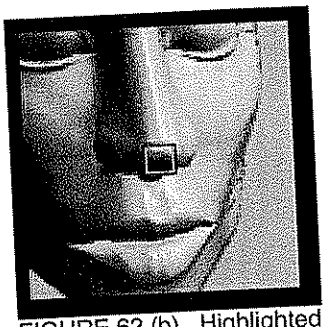


FIGURE 62 (b). Highlighted target image sample. A 20X20 box was displayed in each target image.

5.3.3.5 Procedure

Subjects were given 20 practice trials, selected at random, with replacement, from the set of 486 possible combinations. Subjects could ask questions during, or immediately after, the practice tirals. For each trial, subjects were shown a piece of the background image with the target item highlighted (Figure 63) and a "Next Trial" button (not shown). The target remained on the screen throughout the trial as a reminder. Subjects were instructed to ignore the foreground menu or palette and select a target feature from the background, as shown in the target image sample. Once familiar with the target to find, they pressed a "Next Trial" button which randomly displayed a palette or one of the two text menu superimposed over a randomly chosen background, at a randomly ordered transparency level. This gave subjects time to study each target item prior to commencing each trial. The timer started as soon as this button was pressed and stopped as soon as a mouse click over the background image was detected. Subjects could take short rest breaks whenever necessary. They had to locate and click on the target item within the background image. Sometimes this task was very easy (e.g., highly transparent menus), while other times it was quite difficult (e.g., opaque menus). In the latter case, if they could not see the target, they were instructed to use whatever means they could to make a "best guess". (There was no "Can't See" option in this experiment in order to assess the role of contextual cues.) We observed and later interviewed subjects about their strategies. Response times and distance from the center of the target were recorded for every trial.

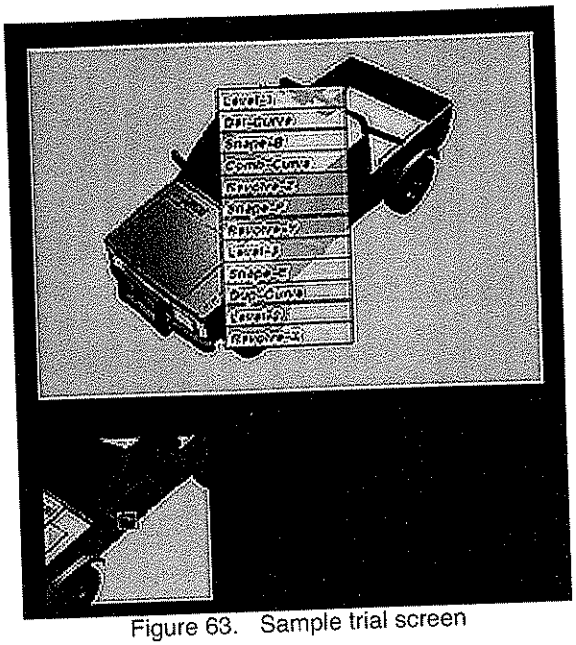


Figure 63. Sample trial screen

5.3.4 Results

We have categorized our results by analysis of both response time and distance for "hits", error analysis of response time and distance, and comments from the interviews with subjects. Recall that a "hit" is defined to be a selection within 20 pixels in any direction of the center of the target. A 9 x 3 x 6 univariate ANOVA was carried out on the selection response time data with Background type (x number of samples), Foreground type, and Transparency as the independent variables. Background types were analyzed by each individual image and by category (i.e., text page, wire frame, solid). A separate 3 x 3 x 6 ANOVA was carried out for the analysis using Background type and is reported below.

Quantitative Statistical Analysis - Response Time for Hits

Statistically significant interaction effects were found for Background x Foreground, Transparency x Foreground, Background x Transparency, and Background x Foreground x Transparency (Table 7). Highly significant main effects (see Table 7) were found for all of our major independent variables: Transparency level (Hypothesis 1), Foreground type (Hypothesis 2), and Background type (Hypothesis 3). In the initial analysis, the background types were not merged i.e., all 3 samples of each type (9 in total) were originally analyzed separately, in case there were unanticipated interaction effects with particular images. We are primarily interested in the interaction effects (as opposed to the main effects), particularly those which involve the Transparency factor.

condition	df	F value	p<
background type	8, 104	19.50	.0001
transparency level	5, 43	13.51	.0001
foreground type	2, 26	7.72	.005
bkgrnd type X foregnd type	16, 208	2.85	.0005
transp. X foregnd type	10, 86	3.12	.001
bkgrnd type X transp	40, 343	2.58	.0001
bkgrnd X foregnd X transp	80, 671	2.13	.0001

TABLE 7. Statistical Results for Main Effects and Interactions

To better understand the interaction effects and to determine if the differences are significant between the individual points plotted within each of the graphs, a Student-Newman-Keuls (SNK) test was run post-hoc as a comparison of means, with an alpha level = 0.05. (This determines the clustering of items within foreground type, background type, and transparency level. and indicates which items are not statistically different.) The results of the post-hoc analyses are described below.

To understand the 3-way interaction effect, it is easier to first present the data on the 2-way interactions and subsequently discusse the Background x Foreground x Transparency interaction. We first examine the Background x Transparency interaction in more detail below (Figure 64).

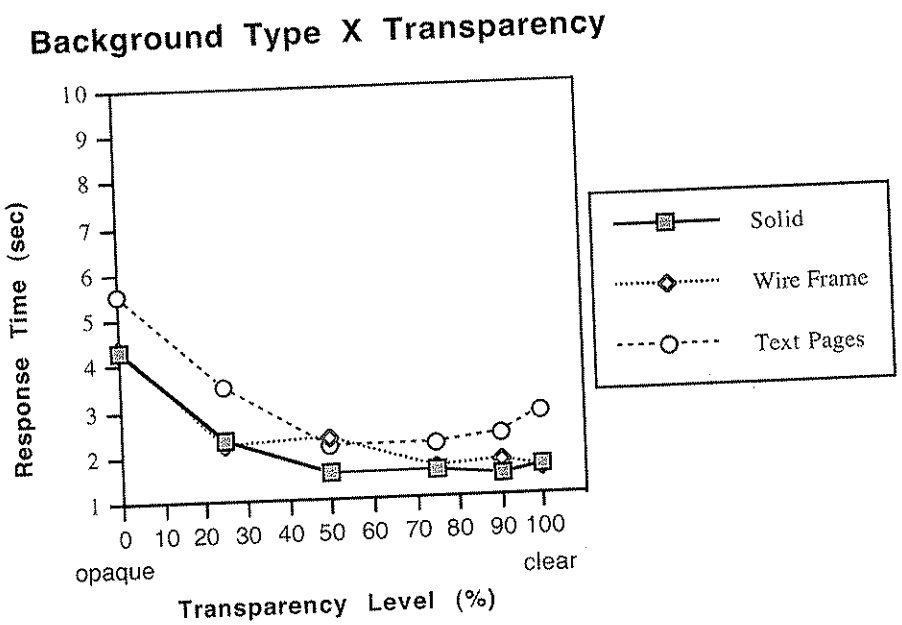


FIGURE 64. Background Type X Transparency Level Interaction

We conducted a post-hoc SNK analysis of background type at *each transparency level* (Figure 64). At 0% (opaque), where subjects were using "best guesses", text pages performed significantly worse than either wire frames or solid images. This was also the case at the 25%

transparency level. At 50%, 75%, 90%, and 100% transparency, the differences between any of the conditions were not statistically significant. At the baseline background only condition (100%), we observed the response time increase (hypothesized to be due to losing the boundaries for the search space). The difference between 90% and 100% however, was not significantly different. This is not surprising since we anticipated only a minor effect, if it existed at all.

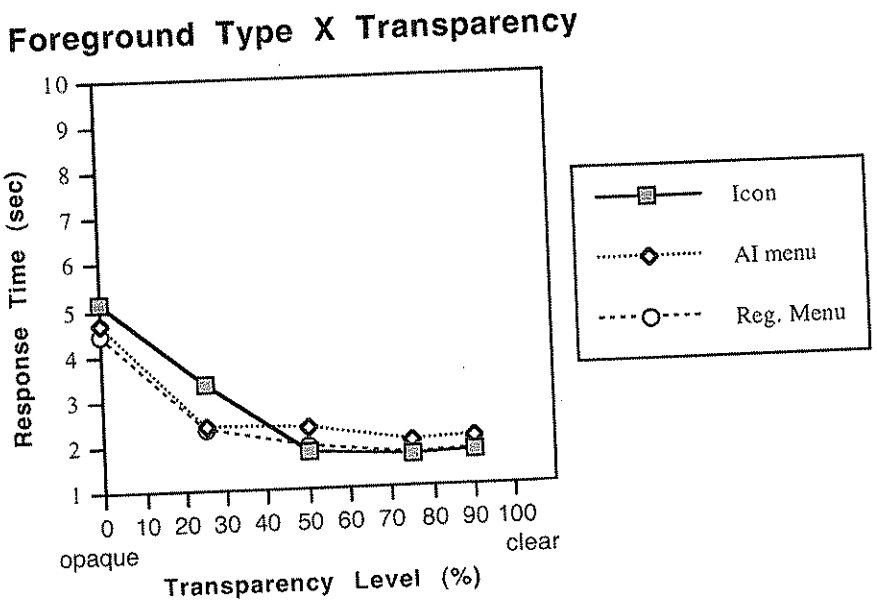


FIGURE 65. Interaction between Foreground Type X Transparency Level

The SNK analysis of the foreground type at each level of transparency ($\alpha = 0.05$) showed a number of surprising effects (Figure 65). At 0% (opaque), all three foreground types showed slightly significant performance differences. This is somewhat surprisingly, particularly since the two text menus are the same shape and similar size. The ANOVA results at 0% transparent showed foreground type as: $F(2, 26) = 4.07, p < 0.02$. (Note that in general we have used an alpha level = .001 as our cut-off point. However, less strict criteria would show this result as significant and hence we report it here for completeness). At 25%, icon palettes performed significantly worse than either AI menus or regular menus. There was no difference between the two menus. (We hypothesized the opposite effect.) At the 25% level, there was a significant interaction between background X foreground $F(16, 211) = 4.11, p<.0001$. At 50%, 75% and 90%, there was no statistical difference in performance between any of the foreground types or between the levels themselves. (At 100% (background only), there are no foreground images and therefore no data is plotted in Figure 65 for that value). We believe that some of the unexpected and complex effects may have resulted from an interaction with the background type, which was not taken into account in the foreground X transparency analysis (i.e., the 3-way

interaction). We further investigated the background X foreground interaction, by running additional SNK post-hoc analyses, reported below.

For each of the foreground types (icon palette, AI menu, regular menu), background was statistically significant at the $p < .0001$ level (Table 7), though the difference between the two menu types was not significant (when collapsed across all transparency levels). The results are plotted in Figure 66. Icon palettes performed worst, regardless of background, though the more disparate the background content, the better the performance. AI menu performed somewhat better and finally regular menu worked best. Text backgrounds had the largest impact in terms of task difficulty, resulting in significantly worse performance. There was no significant difference in performance between wire frame and solid images, regardless of menu type.

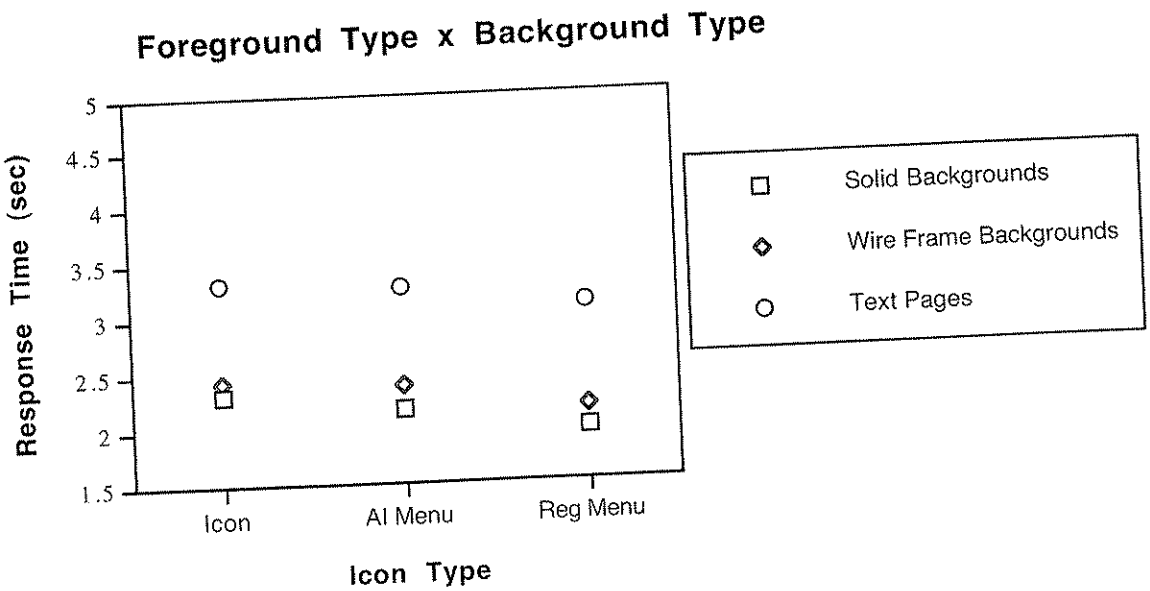


FIGURE 66. Interaction between Foreground Type X Background Type

We can now apply the above analyses of the various 2-way interactions to the interpretation of the 3-way Background x Foreground x Transparency interaction. At transparency levels above 50%, neither the background type, nor the foreground type have any statistically significant impact on performance. Post-hoc SNK analysis showed that performance is not statistically different for 50%, 75%, 90%, or 100%. At levels below 50%, not only is there an overall performance degradation but the extent of this degradation varies based upon the background type, or the foreground type, or both in combination. Solid and wire frame backgrounds have the best performance, followed by text pages. The foreground types showed the exact opposite ordering based on type; icon palettes (i.e., solid images) have the worst performance, followed by the two text menus, which were not different. Overall the background images seems to play a stronger and unequal role in determining the magnitude of the performance differences. In

particulat, text pages have a larger impact on degraded all foreground performance than the other background image types.

The main effects graphs shown below support: Hypothesis 1 (transparency increases result in faster response times, Figure 67a), the order of foreground type predicted in Hypothesis 2 (Figure 67b), and the ordering of background type performance (Hypothesis 3, Figure 67c). However, it appears that we generally underestimated how well wire frame images would perform. Across background and foreground types, post-hoc SNK indicated that there were no statistically significant difference in transparency levels between 25% and 100%. Two statistical groupings resulted: 0% and 25%+50%+75%+90%+100%. Across all transparency levels and foreground types, background types were grouped as expected: solids, wire frames, and text pages. Also across all transparency levels and across all backgrounds, foreground type was surprisingly grouped as: Icon+AI menu, and Regular menu. However, the groupings are much more relevant when analyzed within (and not across) transparency levels.

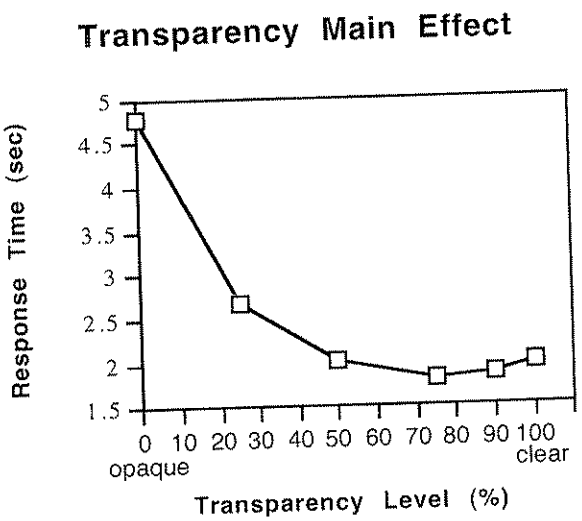


FIGURE 67 (a). Main Effect for Transparency Level across all background types and foreground types

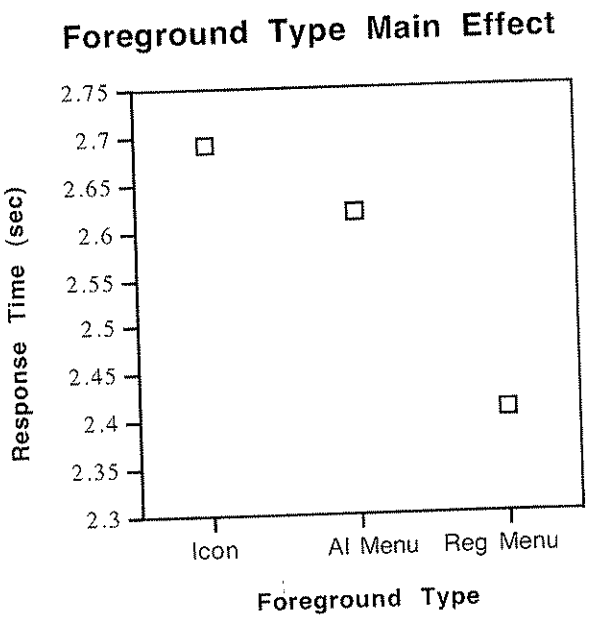


FIGURE 67 (b). Main Effect for Foreground Type across all transparency levels and background types

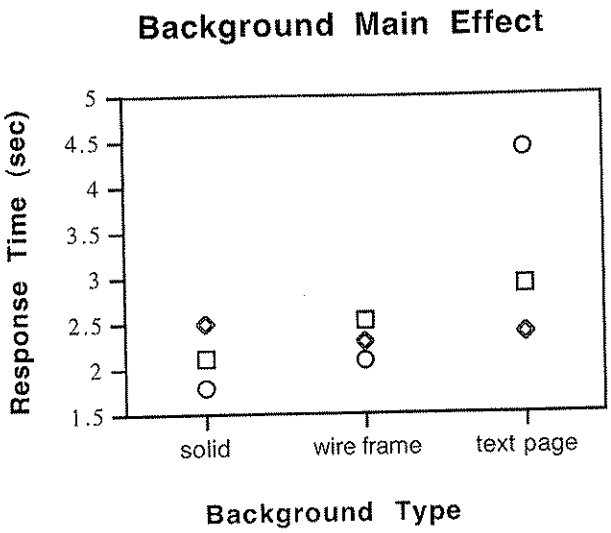


FIGURE 67 (c). Main Effect for Background Type across all transparency levels and foreground types, shows all 3 samples of each background type

Quantitative Statistical Analysis - Distance for Hits

In addition to the analyses conducted on the response time data, we conducted analyses on the distance data. Once again we used post-hoc Student-Newman-Keuls (SNK) tesst as a comparison of means ($\alpha = 0.05$). Recall that distance for hits meant that the subjects had to select a point within 20 pixels of the target center. This section examines distance data for selections which were deemed to be "hits" and is intended to complement the response time data for "hits" presented in the preceding section. Statistically significant interaction effects were obtained for Foreground x Transparency and Background x Transparency. Main effects were found for all three variables (see Table 8 below).

condition	df	F value	p<
background type	8, 88	19.50	.0001
transparency level	5, 43	257.90	.0001
foreground type	2, 26	5.9	.005
foregnd type x transp.	10, 86	3.74	.0005
bkgrnd type x transp	40, 343	2.75	.0001

TABLE 8. Statistical Results for Main Effects and Interactions - Distance

The graph below (Figure 68), shows the mean distance for each level of transparency, per background type (Background x Transparency). Not surprisingly, there is a clear convergence trend as the transparency level increases (i.e., the background target is more visible). Between the clear menu level (100%) and the 50% transparent level, there were no statistical differences between background types. Based on an SNK post-hoc analysis ($\alpha = 0.05$), at 25% transparency,

three grouping resulted: solid, wire-frame, and text pages. However, the performance ordering was the reverse of what we hypothesized. Solid images performed worse in terms of distance than either text or wire frames. Text pages performed the best of the three background types. At 100% the same three statistically significant background type groupings remained (i.e., text, wire frame, and solid).

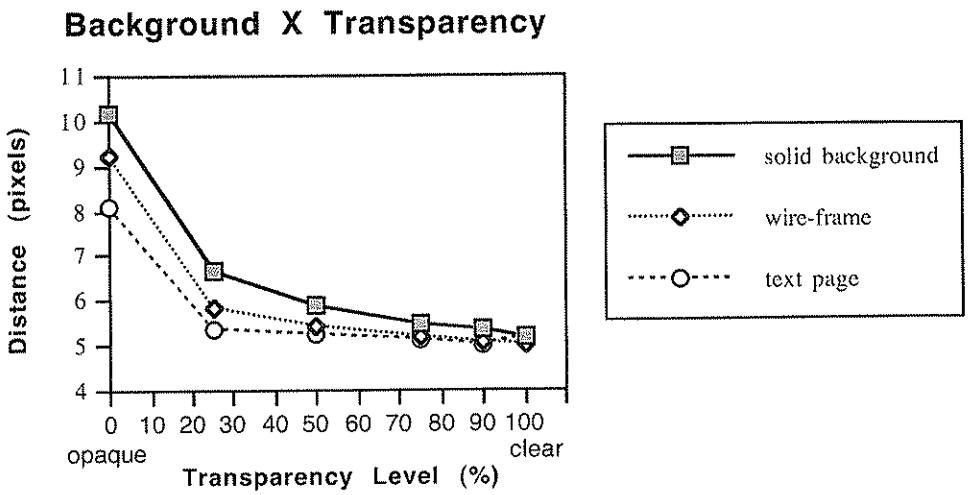


FIGURE 68. Background Type X Transparency Level Interaction

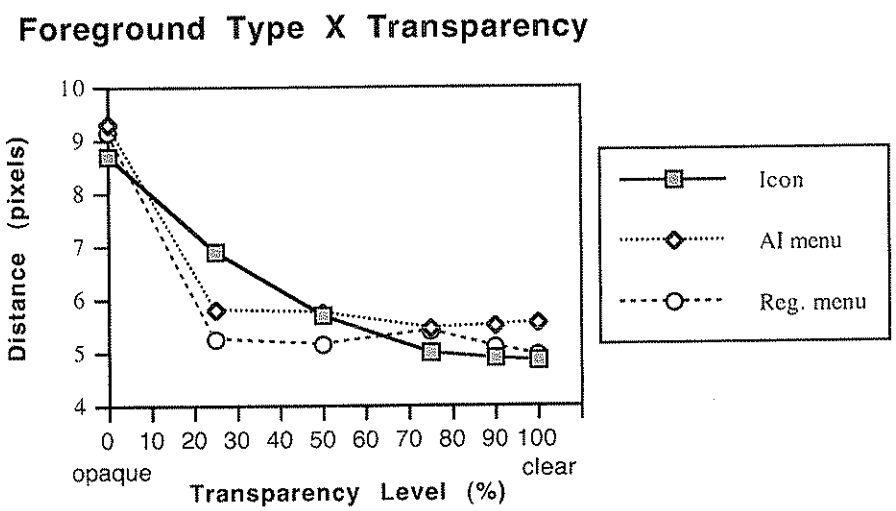


FIGURE 69. Interaction between Foreground Type X Transparency Level

The SNK analysis of Foreground x Transparency showed a number of unexpected effects (Figure 69). As expected, at 0% (opaque) all foreground types performed equivalently in terms of distance off the target. At 25%, the two menus showed a statistically significant difference from the icon palette, but not from each other. They performed significantly more accurately than the icon palette, which we did not anticipate. ANOVA F-tests results at the 25% level were: Foreground type, $F(2, 26) = 13.82$, $p<.0001$. At 50%, 75%, and 90% all foregrounds again

showed no statistically significant difference. At 100%, there were statistically significant differences between the AI menus and the regular menus + icon palettes (which were not different). We believe that some of the unexpected and complex effects may have resulted from an interaction with the background type, which was not taken into account in the foreground X transparency analysis. We present the results for the Foreground x Background analysis below.

An analysis of the Foreground x Background interaction a significant difference between background types only when the foreground was an icon palette (Figure 70). The two text menus did not show significant interactions with Background type. When combined with the above analyses, we find that this significant interaction occurs at the 25% transparency level, where there was a significant interaction between Background x Foreground, $F(16, 192) = 2.91$, $p<.0005$. Our post-hoc analysis further suggests that text pages produce the most accurate results, independent of foreground type. However, they have the largest beneficial effect for icon palettes. Regular menus seem to remove any advantage that the text background have; all backgrounds used under regular menus showed no statistically significant differences.

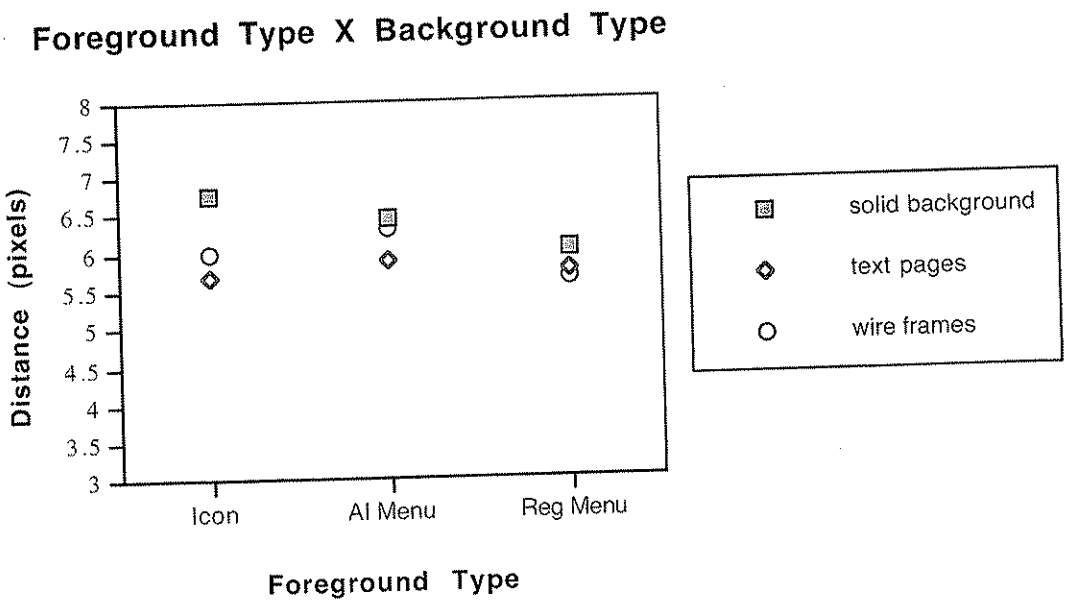


FIGURE 70. Background Type X Foreground Type for distance off target, hits

Targeting Error Results - Response Time for Misses

Error trials were removed from the analysis of the above response time data and were subsequently analyzed separately. In total 10% of the trials resulted in *targeting errors* or misses. Targeting errors are defined to be selection made outside of the highlight box around the

indicated target (i.e., more than 20 pixels away from the target). This corresponds to approximately to 1/2 cm. The error breakdown is shown in Table 9 and Figure 71 below.

transparency level	number of trials
0% - opaque	419
25%	50
50%	32
75%	26
90%	25
100% - clear	17

TABLE 9. Errors due to target misses by substitution

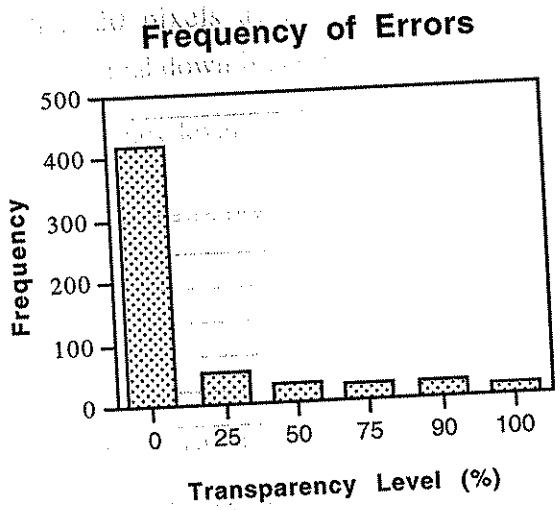


FIGURE 71. Frequency of errors by transparency level

As expected, significantly more errors occurred when the foreground was opaque (0%) than for any other condition. The number errors gradually decreased as the transparency increased. There were no significant differences in error rate amongst the other transparency levels.

An ANOVA was run on the response time data for misses. There were no significant main effects. However, there were several significant interactions: Foreground x Transparency was significant $F(10, 26) = 8.38$, $p<.0001$, as was Background x Foreground $F(16, 108) = 2.60$, $p<.005$. This suggests that the time it takes to respond is determine both by the foreground type and by the combination of foreground to background matching. An SNK post-hoc analysis was conducted to determine the exact nature of the Transparency x Foreground interaction effect. No statistically significant differences were found for the icon palette, across any transparency level. From Figure 72, we can see that icon palettes were fairly resistant to misses, regardless of transparency level. Both menu styles had widely fluctuating response times.

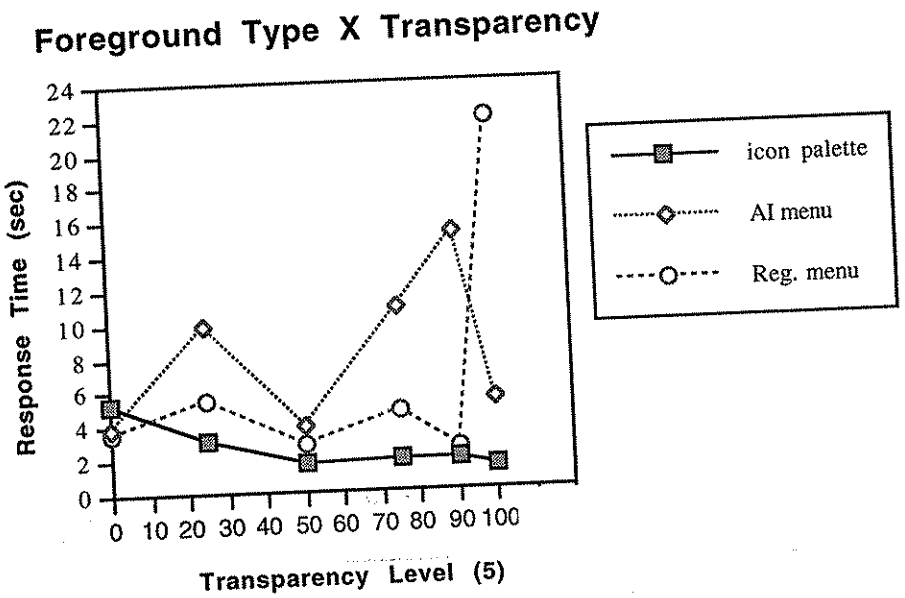


FIGURE 72. Miss response time performance for Foreground X Transparency

Targeting Error Results - Distance for Misses

Lastly, we analyzed the distance data for the misses. These are presented below in Figure 73. The distance data for error trials is perhaps more relevant than those for response times, in terms of determining a typical speed-accuracy trade-off. An ANOVA was conducted on these error data. Significant main effects were found for Transparency $F(5, 46) = 3.45$, $p<.01$, and for Background $F(8, 94) = 3.01$, $p<.005$. A significant interaction was found for Background x Transparency $F(45, 120) = 38.95$, $p<.0001$ and for Background x Foreground $F(16, 108) = 3.74$, $p<.0001$. The variance within each background type was substantial.

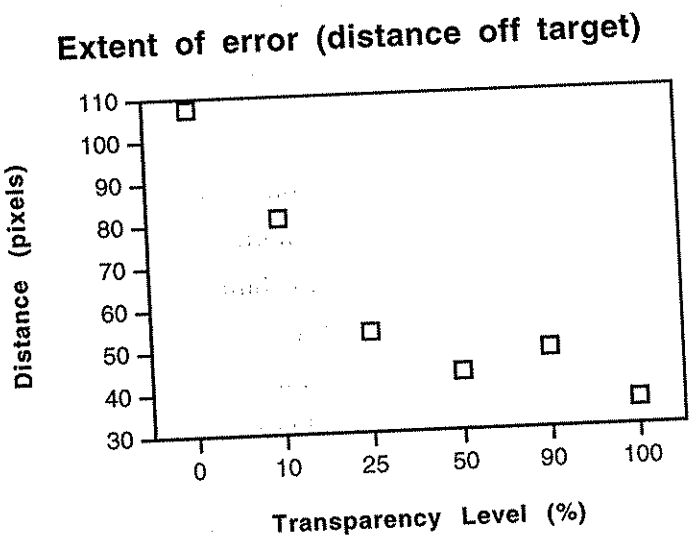


FIGURE 73. Distance as a function of transparency level.

Qualitative Results

Subjects were interviewed after the experiment to assess their perceptions about which conditions were hardest and easiest. Also, they were asked to comment on what they believed their "target locating" strategy to be, particularly under the poor visibility conditions. These comments were augmented with the experimenter observations of subjects' strategies (which were often apparent from the method in which they moved their cursor prior to selection).

In general, most subjects found text backgrounds the most difficult and solid images the easiest. Three of the 14 subjects thought wire frames were hardest. Locational strategies varied slightly but some general trends were apparent. For text page backgrounds, most subjects used adjacent words and paragraph breaks as contextual cues to locate the text target word. In particular, they noted longer words that occurred nearby, since these words were most likely to appear partially outside the overlapped region. From this nearby word, subjects estimated the angle and distance to arrive at the target word. After a number of trials, subjects gradually developed a sense of approximately where the target word was located.

For both wire frame and solid images, subjects commented that they used nearby objects to determine where the target was located. In some cases, they estimated the angle and distance from a nearby and visible feature. In other cases, subjects actually traced along an object edge in both the x and y axis to arrive at the point of intersection. One subject was observed practicing the movement pattern on the highlighted sample target before trying to replicate the same movement on the actual image. For instance, to find the side mirror (rear view mirror) on the truck, many subjects traced along the edge of the windshield (X axis) and combined it with tracing along the car door edge (y axis). This was the most frequently used strategy. Some subjects used color cues to determine where edges were for the solid images. They knew the target was over a "blue area" and so they automatically focused on only the blue areas in the image to reduce the search space. In general, using contextual cues from the image itself was the most popular search strategy, independent of the background type. There were several subjects who employed more unique strategies. One subject used the relationship to the edge of the screen to locate targets within the image. Eventually this subject migrated to using positional information about other items within the image. Two other subjects claim that they use the relationship to the "next trial" button to determine the direction and extent of movement required to hit the target.

Subjects also commented that targets closer to the edge of the menu or icon palette were easier to find and hit accurately. Most subjects commented that the background only condition took longer because they didn't have the menus or palette there to identify where to look. Targets with well-defined edges or boundaries were easier to hit than targets with no clear cut edges (e.g., the truck mirror which was well-defined versus the tip of the nose, where the nose blends seamlessly into the face).

For all background types, eventually subjects became somewhat familiar with the set of targets. Almost all subjects commented that they used "muscle memory" (i.e., they knew approximately the angle and extent of the movement required to arrive at the proper "neighborhood" for a given target).

5.3.5 Discussion

The main effects in response time for hits reflected the general curve shapes that we had hypothesized for transparency, for background type performance, and for foreground type. Transparency did seem to display the slight drop off at 100%, confirming our suspicions that the lack of boundaries on the search space would increase the response time over images which had faint menu or palettes outlines (Figure 67a). Subjects' comments about the background only condition also confirm this.

The main effect for background also supported our hypotheses that text pages would perform the worst and solid images would perform the best (Figure 67a). The post-hoc analysis clearly revealed this to be the case (Figure 64). This tended to match subjects' subjective perceptions. However, we overestimated how poorly wire frame images would perform. These seem to perform almost as well as solid images. In hindsight, this is not surprising. We suspect that the more dense the wire frame image is, the more closely it will approximate the performance of solid images. Also, wire frames reveal a number of contextual cues which provide a more "gestalt" perception of an object. Not surprisingly, the target search for features on an object is significantly different than the search for a word within a text page (which is inherently more linear). This means that wire frames and solid images should be quite similar in terms of target search strategies and performance (which was what the results indicate). Aside from this slight underestimation, our expectations for transparency levels and background types would seem to be supported.

Our analysis of foreground type revealed a much more complex interaction than we had hypothesized. While the general shape of the curve fit with our expectations (Figure 67b),

further analysis revealed some odd interactions (Figure 65, 66). We believed that, like the Icon Palette and Text Menu experiment, the less complex foreground images (i.e., the icon palette) would perform much better than either of the menus. This would be consistent with our observation in the Icon Palette and Text Menu experiments, where solid foregrounds and backgrounds performed better than wire frames, which were equivalent or slightly better than text. In fact, we found the exact opposite for distance or accuracy performance, with respect to background types (i.e., solid backgrounds performed the least accurately, followed by wire frames and then text pages). In the case of foreground types, we expected AI menus to be worst, followed by regular menus, followed by icon palettes. Overall this seemed to be the case, but an analysis by transparency level showed that this ordering was true only for levels of low transparency (close to opaque menus). At levels above 25% transparent, the ordering reversed such that icon palettes performed worst, followed by the two menu styles. One possible reason for this might be due to the size differences between the menus and the palette. The menus are longer and thinner, meaning that reference items which lie outside the overlapped area might be used which are slightly closer to the target item than reference items available outside the palette. The distance analysis results seem to support this explanation (Figure 69). At levels below 50%, icon palettes generally showed larger selection distances off the target.

Error rates fit our predictions reasonably well. Opaque foreground images performed poorly but overall subjects still had less that 10% misses. This implies that contextual cues provide a powerful mechanism for accurate "best guesses" which result in frequent "hits". In this experiment there is no real lower level cut off point. Such a point must be determined by a combination of unacceptably slow response times and unacceptably high error rates. Even at 0%, subjects could select targets 90% of the time. Note that this was based on our criteria for hits and misses, i.e., within 20 pixels in any direction.

5.3.6 Summary

Our original hypotheses that increases in transparency would improve performance up to almost 100% were confirmed. At 100% the slight performance drop-off we hypothesized occurred and subjects' comments indicated that this was indeed due to the unbounded search space (which the occurrence of superimposed menus or palettes provided). Solid images did indeed perform the best and text pages the worst. However, we overestimated how poorly wire frame images would perform. In fact, they were almost as good as the solid images. This was particularly true for denser wire frame images. Search strategies for words differs significantly from strategies used to locate features within an image. This latter case may have contributed further towards performance similarities between solid and wire frame images versus text pages. The foreground

types were generally consistent with our predictions based on the earlier Icon Palette and Text Menu experiments at low levels of transparency (i.e., less than 25%). However, when transparency levels were above 25%, icons performed worse than either menu type. We believe that this was due to the size of the menu items versus the icons and the extent to which the wider icon palettes blocked contextual cues.

5.4 Implications for Design

Combining the results from all of the experiments should provide us with an idea of the trade-offs and potential optimal design points. We have combined the graphical results from the Text menu experiment with the results from the text menu conditions of the Background Selection task (Figure 74). From this figure, if both background and foreground task are given equal weight in importance, then 50% is precisely the optimal trade-off point for regular text menus. For AI font menus this point actually seems to be close to 75%, reflecting the higher interference resistance of this font type. This implies that designs using regular fonts could tolerate 50% transparency without significant loss in performance. Designs using AI fonts (or possibly other outlining techniques) perform quite well at levels up to 75% transparent. Clearly, designers would have to take into account both the "cost" of errors (often determined by the availability of "undo" commands) and the relative importance of selection from the foreground versus selection or awareness of the background. This might alter the weighting factor and hence the optimal location on the curve.

We next combine the results for the icon palette using results from the Icon Palette foreground selection experiment and the icon palette condition of the Background Selection experiment. These are shown in Figure 75. Looking at the worst case icon (line art icons), we can see that again 50% transparency seems to be the optimal trade-off point. At this level, performance is relatively unaffected by the addition of transparency in terms of either background or foreground task selection.

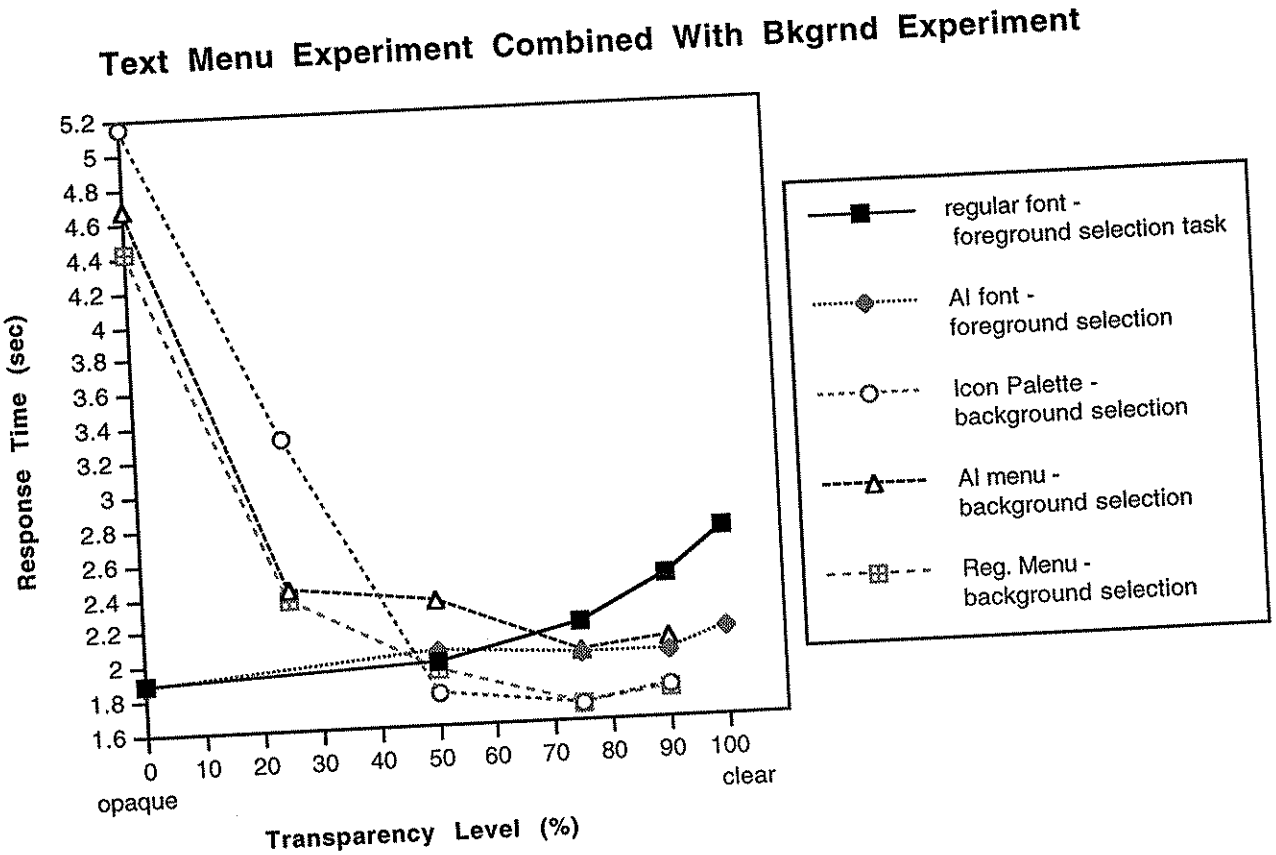


Figure 74. Text Menu results of combined experiments.

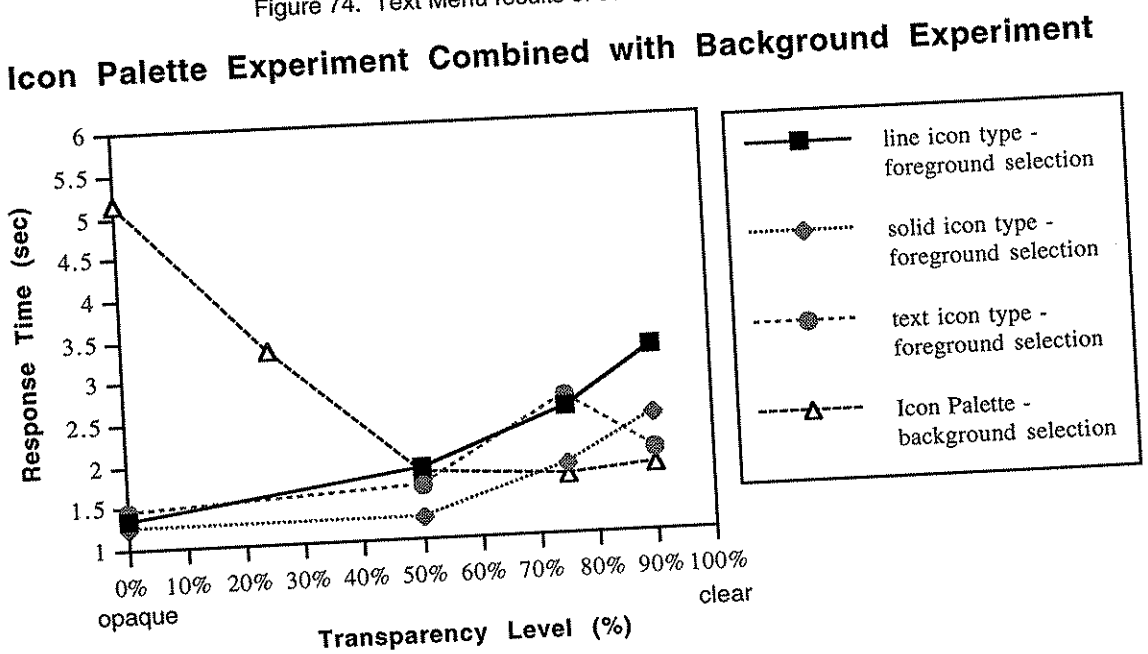


Figure 75. Icon Palette results from combined experiments.

We next combine the results for the icon palette using results from the Icon Palette foreground selection experiment and the icon palette condition of the Background Selection experiment.

These are shown in Figure 75. Looking at the worst case icon (line art icons), we can see that again 50% transparency seems to be the optimal trade-off point. At this level, performance is relatively unaffected by the addition of transparency in terms of either background or foreground task selection.

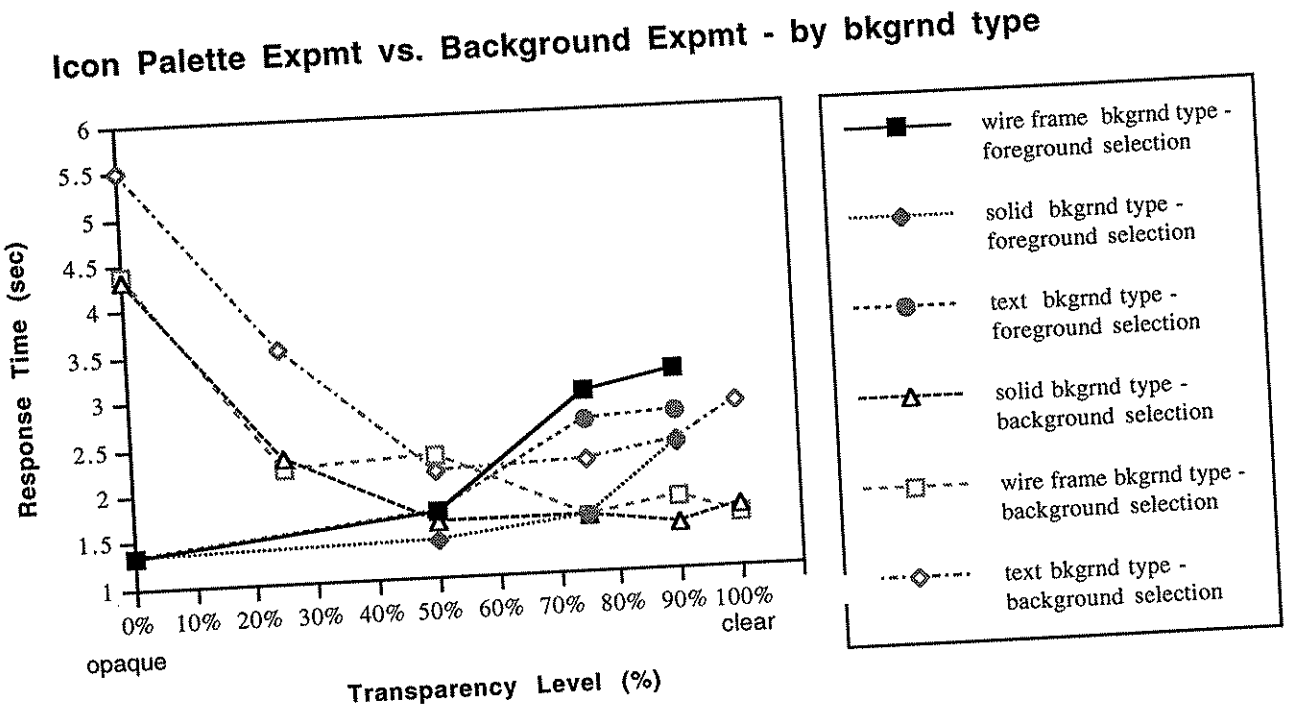


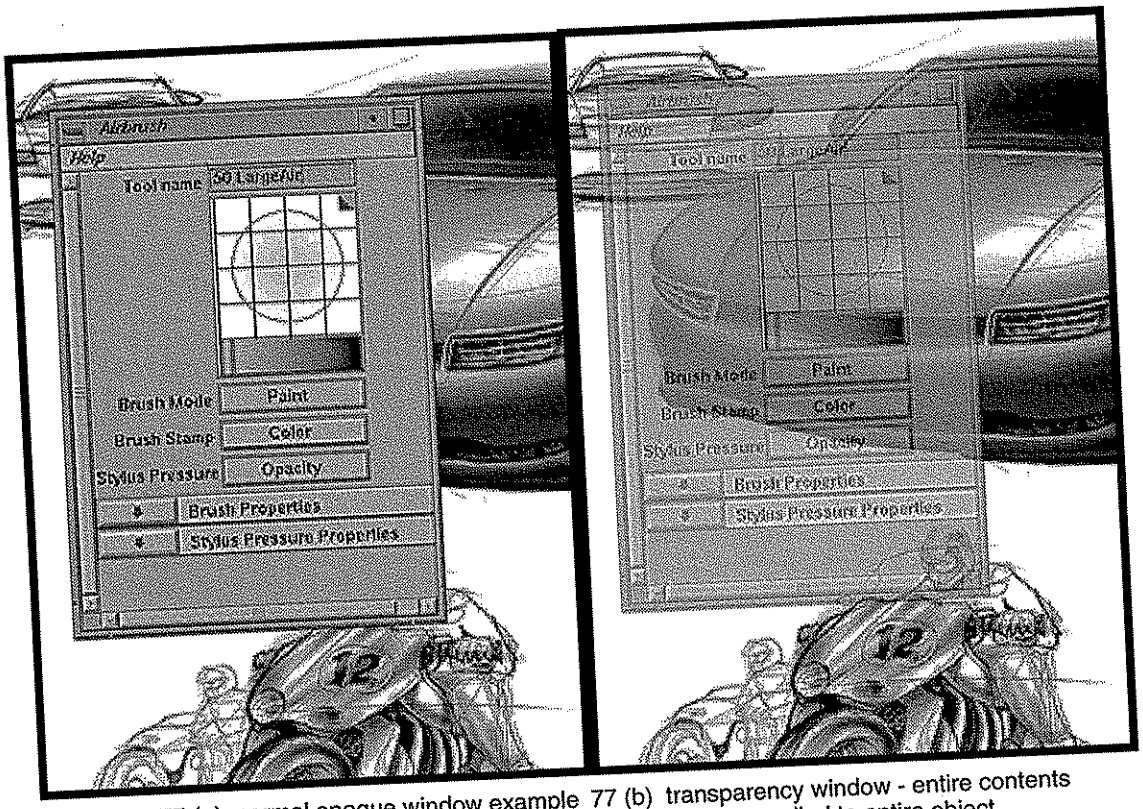
FIGURE 76. Comparison of Background Type across experiments

6.0 Prototyping

In this chapter, we first describe a static prototype comprised of a series of non-interactive screen layouts constructed from our target application domain, a 3-D modeling and paint application (see Figures 77 and 78). Although this prototype was not designed to work with user input in a highly-interactive manner, users discovered that they could move certain objects within the screen layout (similar to moving windows on a screen). This was a useful artifact of the tools we used to build the prototype. Other than this, the prototype did not recognize user input and hence we considered it "static". In addition to the static prototype, we built an interactive prototype which combined a very simple graphical drawing program with transparent images and menus. The purpose of this prototype was two-fold. First, it allowed us to dynamically change transparency levels and obtain informal user feedback within the context of a small application. Second, it allowed us to better assess the implementation alternatives and challenges within a small, controlled body of code. This in turn permitted us to make implementation decisions and determine trade-offs for the later system used in the case study, described in Chpater 7. Some of the prototype development occured in parallel with the experimental design described in Chapter 5. As such we were able to quickly and informally test out AI outlines and other visuals prior to formally collecting data. In this chapter, we summarize the user feedback and what we learned based on the development of these prototypes.

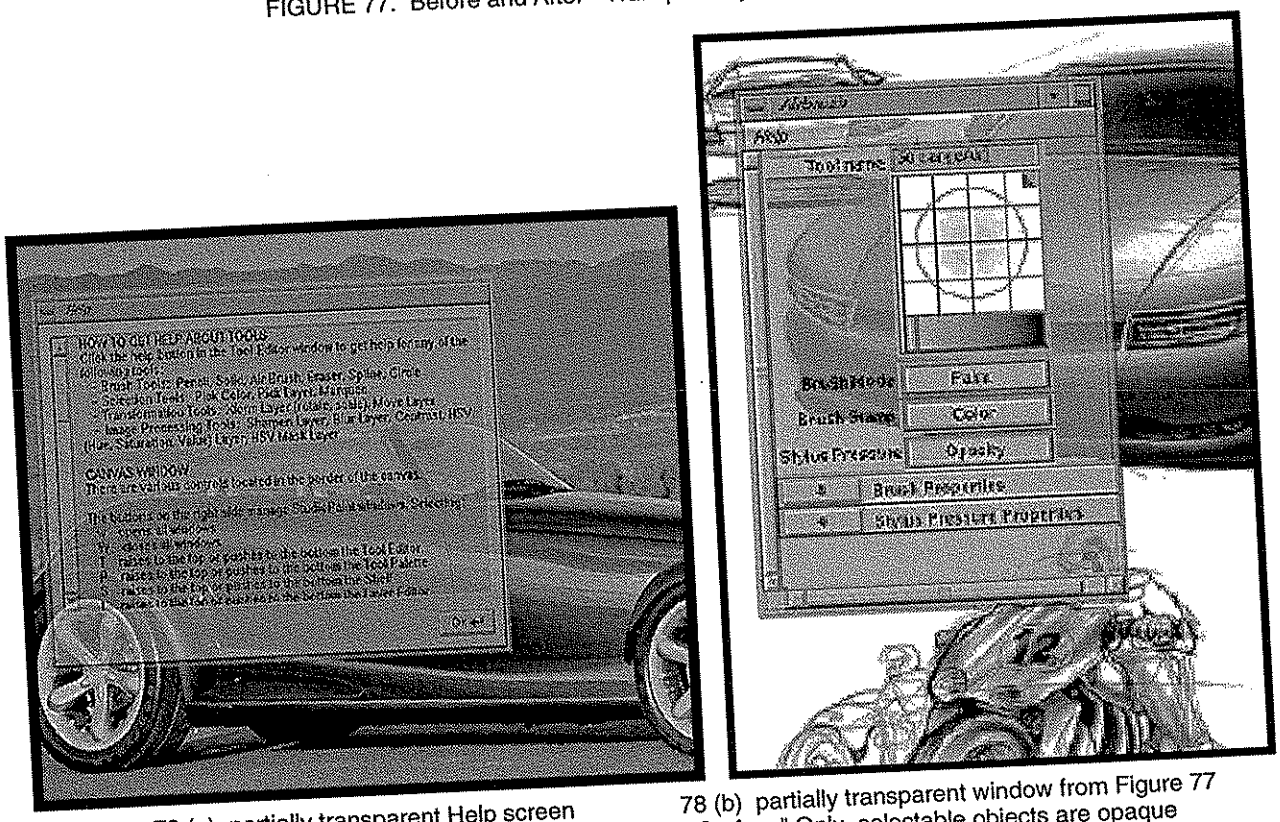
6.1 Interface Alternatives for Transparency

Within our proposed depth multiplexing strategy (see Chapter 2), there are several different types of designs which achieve a transparency effect. In the first method, the entire contents of the interface object can be made transparent (Figure 77b). This includes not only the "surface area" (as shown below) but also the text labels, borders of items, icon images, and button areas. Surface area is roughly equivalent to the non-selectable regions of traditional user interface windows. In effect, the *entire* contents of a menu or window are made see-through. In the second strategy, *only* the surface area of the window is made transparent (Figures 78a and 78b). All text labels, buttons, borders and icons are left opaque to maximize legibility. This approach has the potential of improving foreground target accuracy but at the expense of potentially obscuring background target accuracy.



77 (a) normal opaque window example 77 (b) transparency window - entire contents

FIGURE 77. Before and After - Transparency is applied to entire object



78 (a) partially transparent Help screen 78 (b) partially transparent window from Figure 77

FIGURE 78. Partially Transparent Interfaces - "Surface" Only, selectable objects are opaque

There are several criteria which suggest why designers would favor one method over the other. The "surface strategy" (e.g., Figures 78a and b) works well in cases where there is a lot of surface area relative to text, buttons, and icons (e.g., dialog boxes and menus are examples of this). This strategy has the advantage of preserving a high degree of legibility. However, there are a number of cases where user interface objects do not have much "surface area" and therefore applying surface transparency would be of minimal utility (e.g., iconic tool palettes and dense interactive windows). In such cases, the entire interface object, including some or all of the content details, would be made transparent. This preserves an awareness of underlying layers at the expense of legibility of text label and distinctiveness of buttons or icons. Clearly, a hybrid combination of these two strategies is possible. For example, we might keep object borders opaque to potentially improve targeting accuracy while making the interior contents and/or text labels transparent. In this case, location or positional cues become important for object identification.

Another method of achieving and applying transparency is to use a context-sensitive interface. In this case the original appearance of the menu or window, for example, is highly transparent. However, the level of transparency depends upon the proximity of the cursor. As the cursor moves over selectable objects these objects become more opaque, thereby improving legibility and thus, targeting accuracy. This strategy provides cues about which targets are currently applicable in the same manner as some highlighting techniques. The assumption is that cursor position and movement are related to selecting items (as opposed to hot keys or power commands). As the cursor is moved over an item, it indicates a potential intent to choose that item, and so, the item becomes more opaque. This dynamic application of transparency could be applied to entire windows, menus or dialogs, or it could be applied to single selectable objects within these larger regions.

In addition to a choice of applying transparency or not, we can also vary the level of transparency within an object or surface area. For example, we might use graduated transparency levels across a window, where higher levels of opacity are used over locations that have higher information density. Presumably, as information in either layer gets more complex (i.e., higher density), the opacity of the foreground layer would correspondingly increase to improve visibility and selectability of foreground objects, again at the potential expense of background visibility. A similar graduated technique is sometimes used in broadcast television (see Figure 79) although it is not context sensitive (since there are no cursors, user interaction is not possible), nor is it based on any analysis of display information. Most television effects are determined by designers, based on aesthetics, prior to the broadcast and only the text is changed over time. However, such imagery is suggestive of more sophisticated interactive applications of transparency which are

sensitive to either user intentions (e.g., cursor position) or to the contents and complexity of the object.

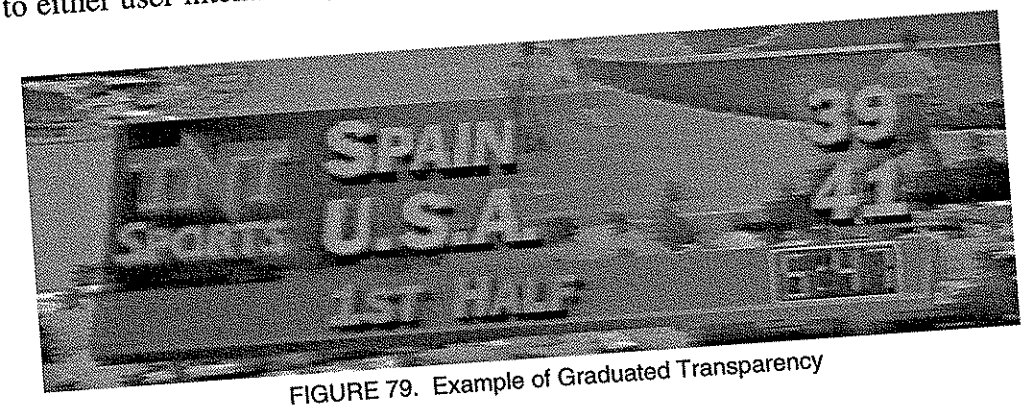


FIGURE 79. Example of Graduated Transparency

6.2 Static Prototyping

In order to gain a better understanding of transparent user interfaces in realistic settings, we created a number of static prototypes based on realistic task contexts within working applications. The prototypes were created so that the level of transparency could be varied for each evaluation. Comparison were made between levels of transparency within an image and across different images. These static prototypes were shown in a series to a variety of users to elicit feedback and comments.

6.2.1 Description

We chose two application domains characterized by tasks requiring a large work or data area which the users manipulated extensively with interactive windows and dialog boxes. These target applications were a drawing/painting system and a sophisticated 3-D modeling/animation system. Characteristics of this task domain are described in Chapter 1.

In the *3-D modeling/animation* system, users need to see a potentially large model (full screen, background) while changing various attributes of the modeling tools (using windows in the foreground), resulting in constant attentional switches. Typically, users might have 3 or 4 such interactive dialog windows open at any given time. They may be changing attributes such as lighting and shadows, surface textures, model shape, motion paths, and extent of animated movements. Similarly, in the *painting* program, users typically work with large in-progress drawings or images, while changing the brush tools, colors and size of effects through palettes and interactive dialog boxes. Again, in the painting application, users have an icon palette and several windows open at any given moment in time. In both cases, work is characterized by focusing on the large background model or image, followed by focusing on the interface tools and windows which are superimposed over this background. Often, users move around the

windows and palettes without making any selections, to reveal an obscured part of the underlying image. The work flow is characterized by frequent attention switches between the interface manipulation tools and the work-in-progress. Overall, in our terminology, such a work process would not be considered fluent.

We generated realistic work areas by loading in partially completed (i.e., work in progress) image files or models and saving them as high resolution screen snapshots. This provided us with realistic background content. We then had expert users start a typical work session and save high–resolution snapshots at various times which depicted the user interface tools and windows which the users had open. This created the foreground content. Using Alias StudioPaint™, which supports merging multiple images as layers, we loaded the background content as one layer, and each of the user interface window images as other layers. Combining these layers (and using some clipping and masking techniques) allowed us to simulate the current look of a painting session, albeit taken statically at one moment in time. We further enhanced our prototype by taking advantage of the blending function of StudioPaint, which supports altering the transparency of entire layers. By changing the settings of the top-most painting layer, which contained the user interface (UI) windows and tools we had imported, we were able to simulate variably transparent user interfaces. In this way, we were able to re-create a number of visually dissimilar but realistic images that users could evaluate (Figures 70 through 85). This allowed us to obtain user feedback in a quick albeit informal manner.

In addition to the two-layer static prototypes (background image and foreground UI tool windows), we also wished to create more complex variations of the UI tool windows. For example, in Section 6.1 we described a design approach which takes advantage of "dead space" or surface area in a window. To test this variation of user interface tools, we created *hybrid* transparent windows (where the text, buttons, and borders are opaque and only the underlying window surface area is affected by transparency levels) (e.g., Figure 82 versus Figure 83). This was accomplished by re-creating each of the graphical objects, text labels, and borders in a separate layer from the original window layer, and aligning them with the window. The window layer was set to be partially transparent, affecting all of the items within it, including the surface area. The new layer containing all of the window objects was carefully aligned to the window layer and was left opaque. Both the text/button/border layer and the interface window layer were then superimposed over the work area layer. This created a visual appearance of having a hybrid interface window (partially transparent and partially opaque), without having to create a costly working system.

Finally, we created transparent icon palettes using the existing product icons, again taking advantage of the StudioPaint layers (Figure 84). The icon palette was imported into a layer and superimposed over various background content, varying the transparency of the icon palette layer. Since the icons were solid rendered objects, there was no surface area per se. Therefore, the entire icon image was made transparent. We felt that this reflected the way designers would create transparent palettes in working applications. The icons were generally designed as tiny line art style images (Figure 84a), or as fully rendered, color images (Figure 84b). The target size is generally quite small for each item within the palette. In fact, the icon palette shown in Figure 84b is the largest size of three possible palette size settings. (In the working product, users almost immediately change this default setting to the minimum size possible, just to minimize the amount of obstruction it causes.)

Sufficiently realistic images were created such that users initially believed them to be integrated into working products, viewed at one instant in time. We had several users of varying levels of expertise evaluate the transparent simulated user interfaces. In effect, we flipped between multiple StudioPaint documents, where a document contained a simulated transparent user interface. The evaluation was a relatively fast, informal method of gaining insights into our next set of experimental variables and into design parameters and design issues. It also had the important advantage of not requiring extensive implementation or coding.

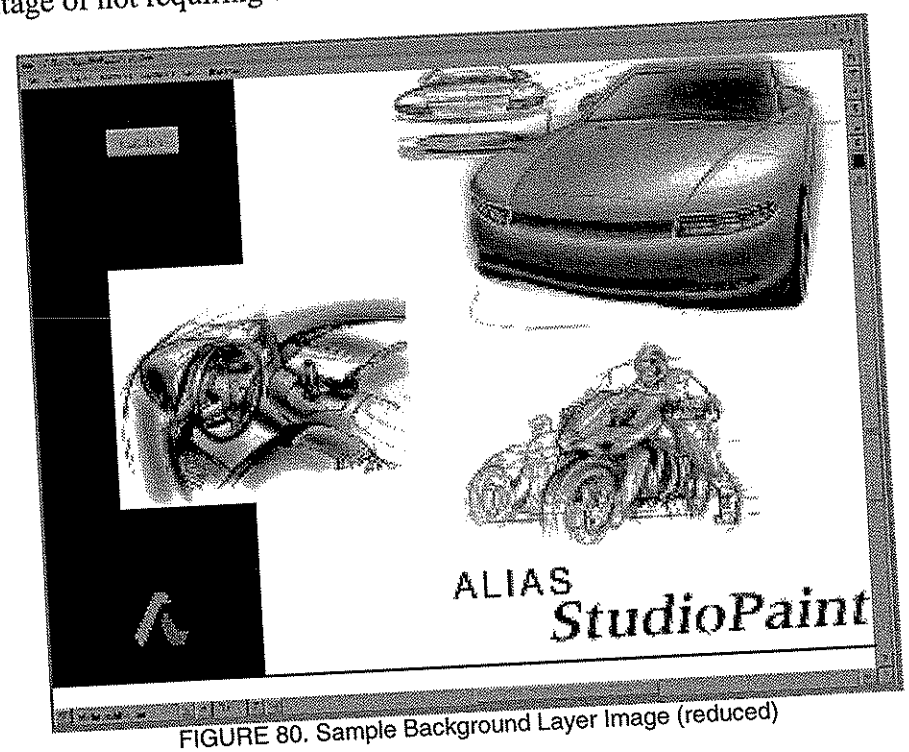


FIGURE 80. Sample Background Layer Image (reduced)

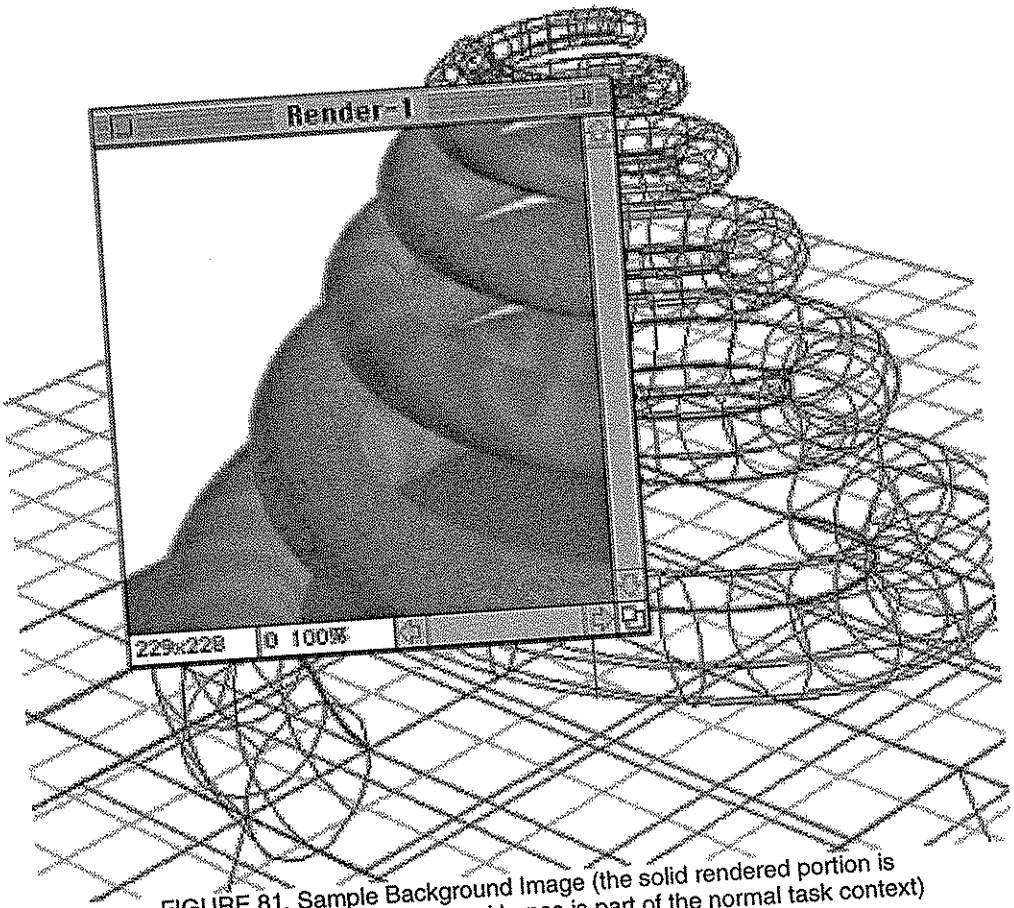


FIGURE 81. Sample Background Image (the solid rendered portion is part of the background image and hence is part of the normal task context)

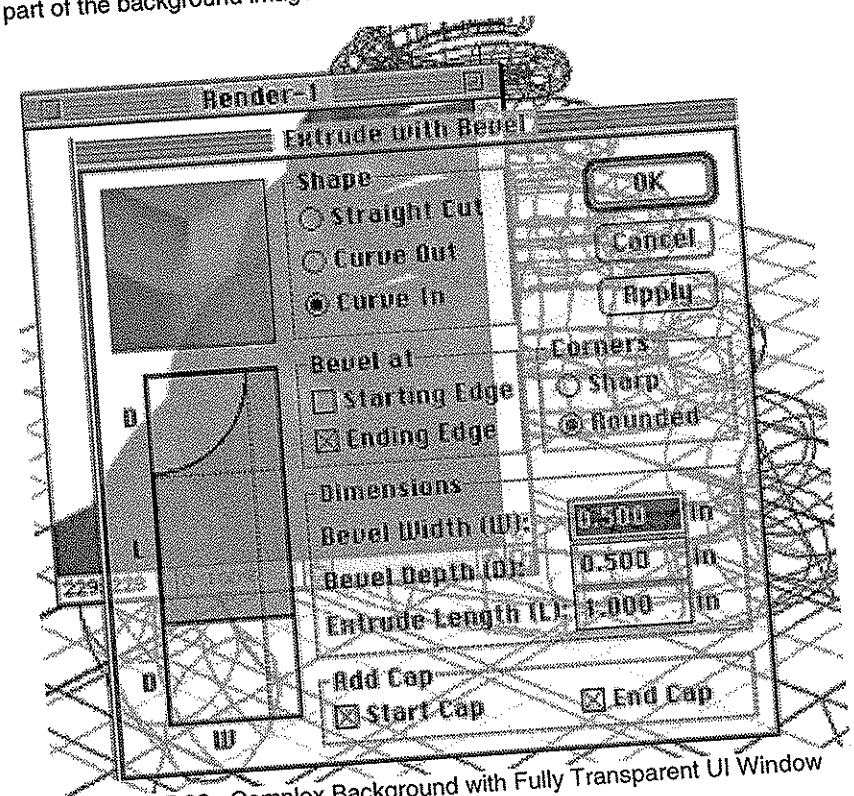


FIGURE 82. Complex Background with Fully Transparent UI Window

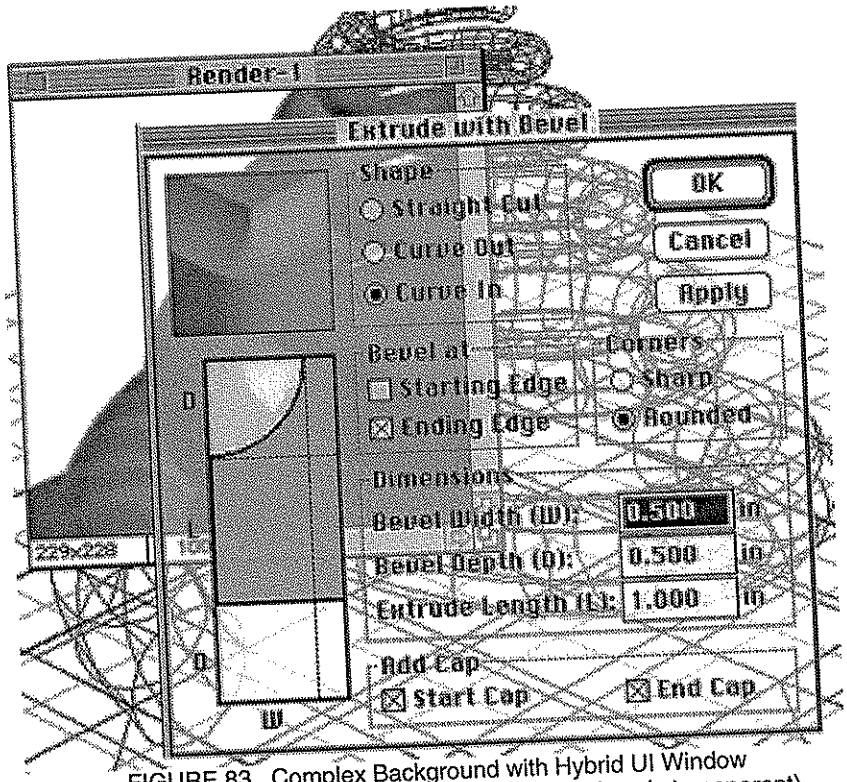


FIGURE 83. Complex Background with Hybrid UI Window
(some objects in window are opaque, window surface is transparent)

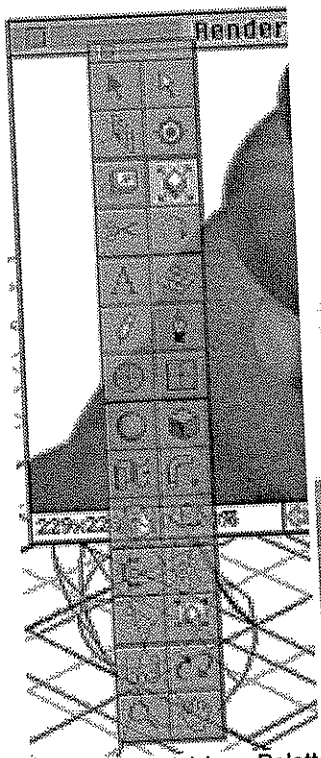


FIGURE 84. (a) Icon Palette

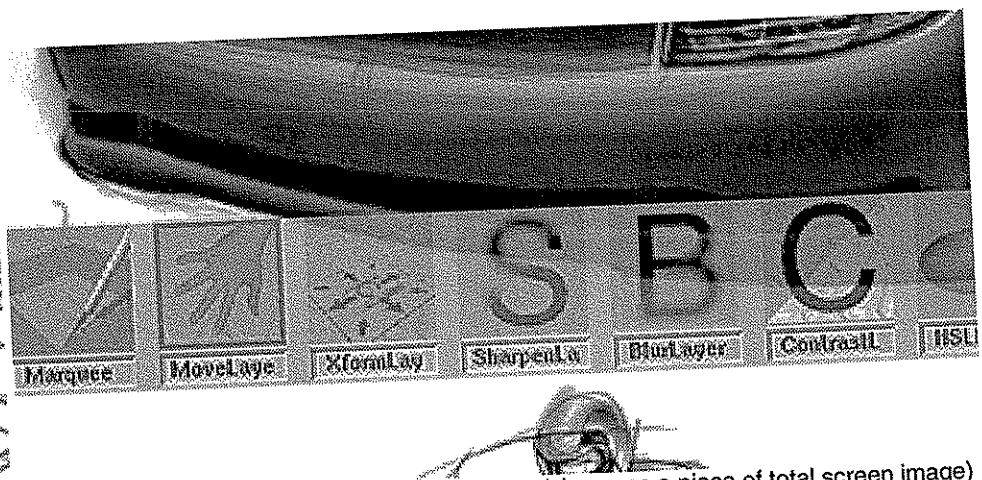


FIGURE 84 (b) Sample Icon Palette (shown as a piece of total screen image)

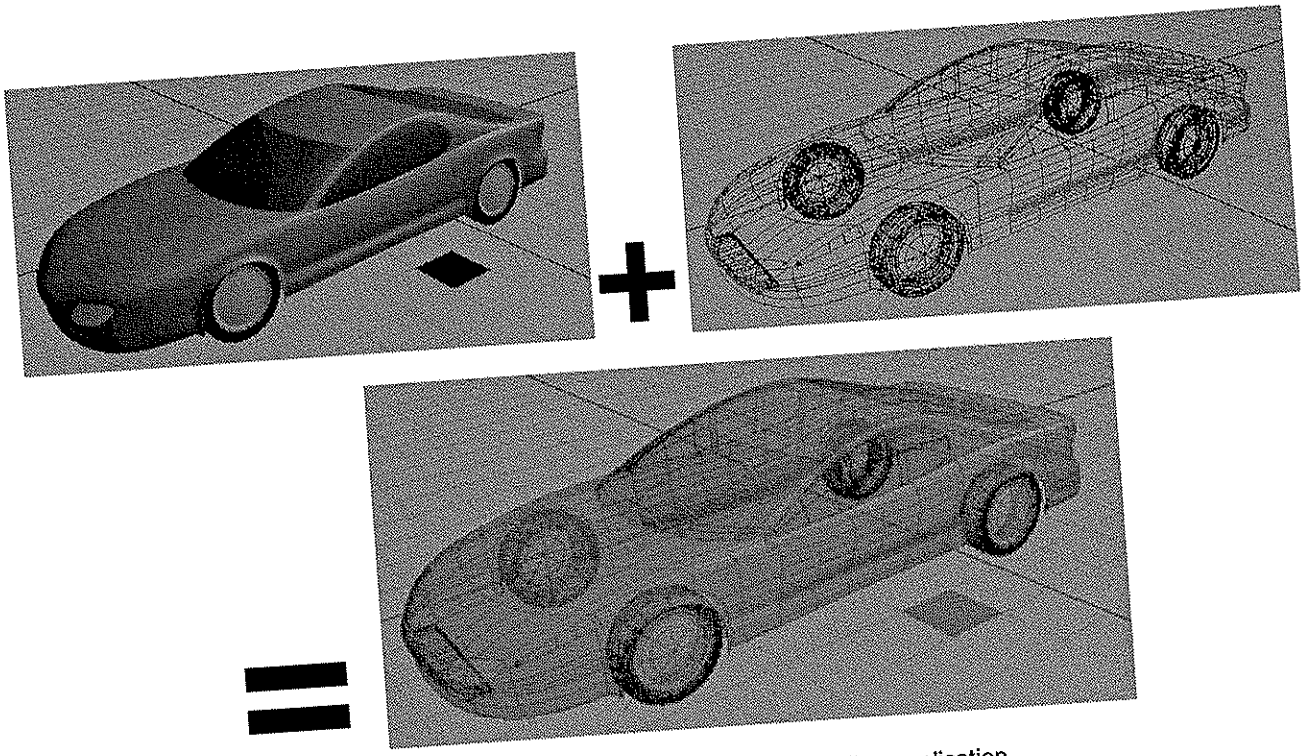


FIGURE 85. Sample of combined images in modeling application

6.2.2 Observations

The degree of visual distinctiveness or dissimilarity between the foreground and background content strongly influences the extent of possible interference and the perceived difficulty of identifying elements. Users found transparent windows (with text, buttons) were easier to use over solid models or images than those superimposed over wire frame drawings. Higher levels of opacity seemed to partially compensate for this in the more difficult task situation (by minimizing interference, as in the Stroop experiment). This finding suggests that the level of detail, or information density, might be a determining factor when choosing transparency levels. (There are a number of possible mathematical methods we could use to calculate the information density of an image. These are described further in the Future Research section). In cases where users were shown highly detailed images (or information dense images), they tended to first move physically closer to the screen to examine the image. When this did not help in discriminating between the two layers, users then opted for decreasing the transparency level.

Users did not like the entirely transparent windows, where all surface area and objects were displayed at the same level of transparency. They strongly preferred the hybrid designs which maintained higher opacity for selectable objects and text. One reason for this was to facilitate determining which objects were selectable at any given moment. Users felt that transparent but selectable objects conveyed a confused message since they had been previously exposed to the

concept that grayed out items were not selectable; transparency and grayed out were too conceptually and visually similar. Users also wondered if they would still be able to even see grayed out options. Opaque text and opaque objects (or at least close to opaque) seem to address these concerns. The hybrid design also had the additional advantage that it preserved visibility for targets while removing the more distracting and interfering elements, not needed by expert users. Expert users claimed that they knew the locations of items and did not need to see the text labels. They only needed targeting information, such as button borders and slider locations.

As familiarity with the interactive window layout improved, users preferred correspondingly increased levels of transparency. They preferred to see "less" of the interactive dialog boxes and more of the underlying image. The dialog box items were needed only as outlines to target selections – the actual legibility of the text was substantially less important. This finding suggests that the border of windows and buttons might be made more opaque than the labels for those objects. It also suggests new and intriguing possibilities for dynamically evolving interfaces based on increased expertise. Finally, these window changes could be based on the task context. For example, users who are actively drawing or painting might wish to continue their strokes "through" an obstructing UI window. If that window became highly transparent as the cursor approached, this time because the combination of cursor position and task context were applied, users would not have to interrupt their drawing stroke to perform window movement operations. All of these suggestions arose as a result of users evaluating the static prototype and making inferences about what it meant to the execution of the task. In particular, users pointed out potential problems and challenges which, in turn, helped to generate some of the design solutions proposed above.

Users had trouble with the gray colored icons over any text or wire frame backgrounds. There are very few cues to differentiate one icon from another, and these cues are subtle differences in one pixel wide lines. Expert users claimed that they used positional cues to choose their tools from these palettes, even when the palettes were opaque. It was difficult to determine how much the transparency would therefore impact tool palette selection, since the spatial cues compensate to such a large extent. The colored icons appeared to be much more interference resistant. Users could still identify them at high levels of transparency over complex backgrounds. Part of this is almost certainly still due to the positional cues, but there is also more information to discriminate one icon from another. For example, users can rely, to some extent, upon shape and color to assist them in identifying icons in poor visibility conditions. Most of these icons have substantial differences in appearance. Unlike the line art style icons, the solid rendered image icons worked well over text or wire frames and worked poorly over solid images, particularly those of similar

color. This supported our suspicions about interference as it related to visual dissimilarity; the more similar items are, the higher the resulting interference.

Finally, a number of users also applied motion to the transparent layer to determine what belonged in the foreground and what was part of the background. (This ability to move layers within our otherwise non-interactive prototype was a beneficial artifact of using StudioPaint as a prototyping tool.) This motion seemed to facilitate reading items from the top layer, despite reading a moving object. Presumably, this was possible since the motion consisted of slow, small scale movements. The fact that users tried this is not surprising, since motion is a particularly strong perceptual cue. Users simply assumed that they could move windows and palettes around and so they tried it out. In fact, we had to make sure the prototype was set up such that the paint program was in "grab/move" mode and the action was set to effect only the UI window layer (as opposed to being in some painting mode or set to another layer or multiple layers, for example). When properly set-up in advance, the move operation did appear exactly like a window move. Users selected the UI window title bar and jiggled the window around to see which items were part of the window and which were part of the background. This showed us how powerful even a static prototype could be when it is built using appropriate images.

6.3 Interactive Prototype[1]

In addition to the static prototype, we wanted a more interactive system to evaluate both the technical feasibility of transparency and to obtain user feedback. To this end, we created a prototype software system, running on a Silicon Graphics machine. This software system, written in C, GL and OpenGL, allowed us to evaluate transparency when using menus, windows, and images. We could interactively vary the transparency levels in a continuous way (using alpha blending).

In using an interactive prototype system, we can control the resulting complexity of the implementation and size of the code, while integrating both measurement tools (to evaluate performance) and more realistic tasks than those necessary for tightly controlled experiments. This simple implementation environment allowed us to quickly evaluate the effectiveness of several proposed design alternatives. The prototyping process provided us with the opportunity to observe the behavior of transparent interface objects in various scenarios, to explore the design

---

1 The interactive prototype development work was done in collaboration with Shumin Zhai, under a service contract to Alias Research Inc., patents pending.

space of these objects, such as the degree of transparency and the design of fonts, and to further hypothesize research issues for formal experimentation. Equally important, prototyping in such a controlled software environment allowed us to assess the technical complexity of generating variably transparent images. These techniques were later applied in designing and coding for our case study application (described in Chapter 7).

Unlike the static prototype, which emphasized the design of graphical objects within windows and icon palettes, the interactive prototype explored issues related to the use and design of text in transparent applications. Text is an integral part of most interactive dialogs and windows and is certainly the key component of linear and radial menus.

6.3.1 Description

Our prototype is a simple color drawing program which contained elemental functions that are often found in end-user software packages. Text, lines, circles, blocks and other simple geometric forms could be drawn either opaquely or semi-transparently. A variety of colors could be applied to any geometric shape or text item. Image files could be imported or exported. Pull-down and pop-up menus were incorporated and the level of transparency of these was adjustable (Figures 86, 87). We also had a text-based help screen which could be set to varying levels of transparency. Font styles, sizes, and colors could be set. Using this fairly simple set of primitives, we were able to generate and evaluate numerous foreground-background combinations which reflect potential design problems that transparent interfaces would have to address. A number of design alternatives were subsequently developed, implemented, and informally evaluated, to improve legibility of text in transparent applications. (These design ideas were tested more rigorously in the experiments described in Chapter 5.)

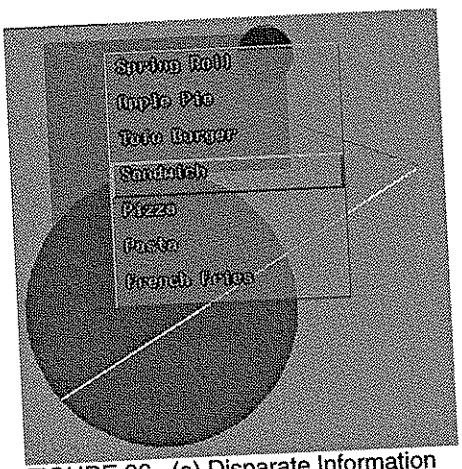
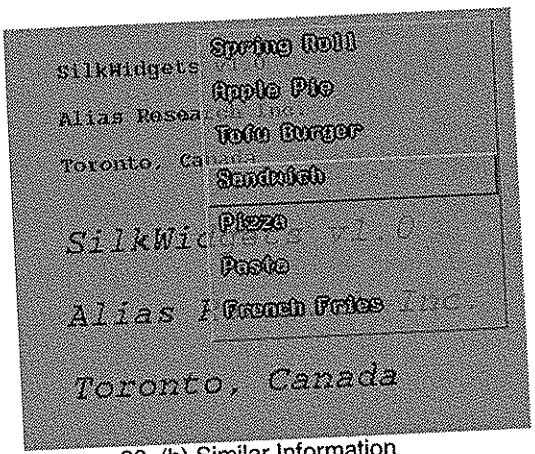


FIGURE 86. (a) Disparate Information



86 (b) Similar Information

Using our interactive prototype, we tested both matched and mismatched foreground and background text, in terms of font style, color, and size (Figure 86b). We compared various colors of text over differently colored backgrounds, varying the transparency of the surface area portion of the menus and help screens. Text was tested over text, over solid objects, and over complex multi-colored images. We built and tested a variety of outline font designs. At each stage, we informally evaluated the legibility of the foreground text and background text (if present). The results of this informal evaluation are reported below. As mentioned, this informal evaluation helped us to determine a number of issues which were reflected in the experimental designs of Chapter 5.

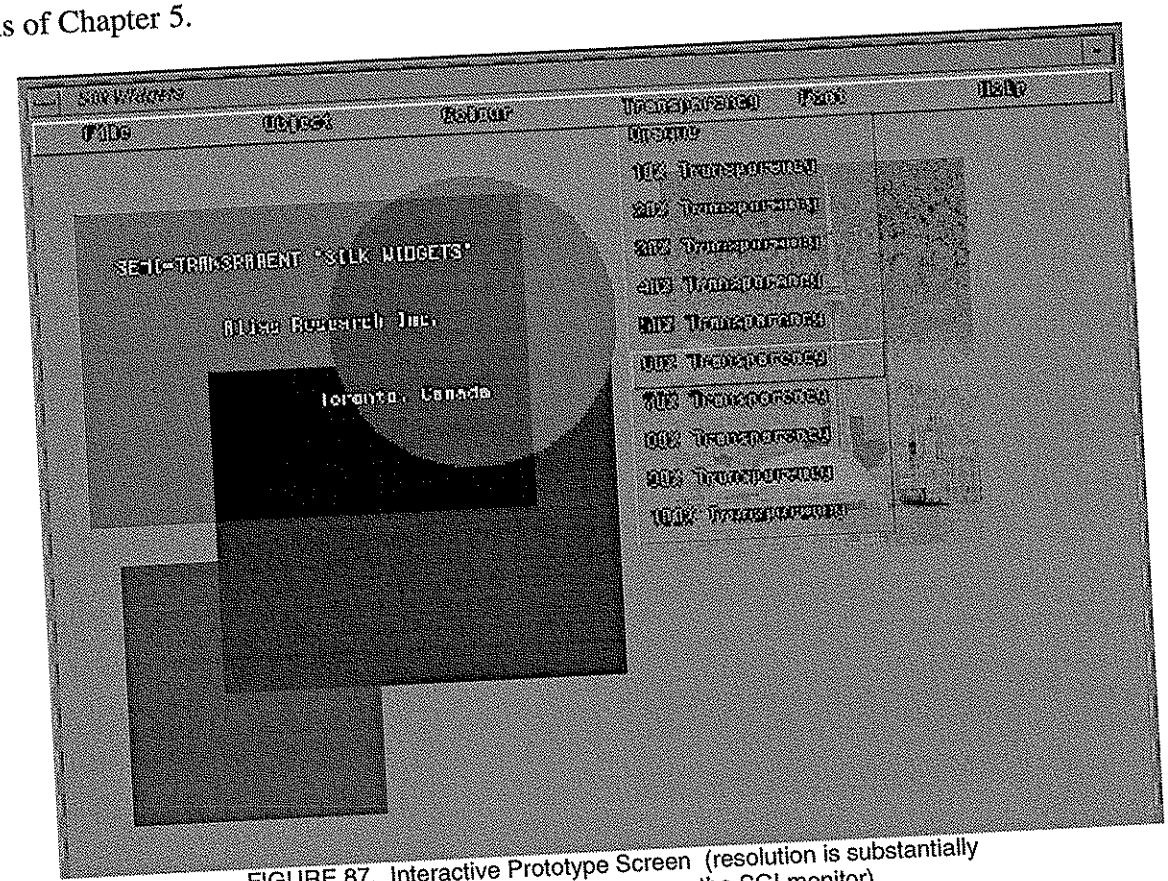


FIGURE 87. Interactive Prototype Screen (resolution is substantially degraded from what appears on the SGI monitor)

6.3.2 Observations

Many of our observations from prototype usage are in agreement with our Stroop Experiment results and our proposed hypotheses based on the attention literature. For instance, when two sources of information are distant in coding format, e.g. the foreground is textual (menu) and the background is graphics, users have little difficulty in focusing attention on the foreground information even though the interface menus are highly transparent and the background content is clearly visible (e.g., Figure 86a). (Note that this assumes visual interference is minimized.

However, it is highly possible to have semantic interference in such situations as is evident from the Stroop experiments described in Chapter 4.) In contrast, when both the foreground and the background have textual information (e.g., menus over text), it is more difficult to focus attention on one source and ignore the interference from the other. There are a number of solutions when this latter case occurs. Differences in font, color, and size improves the separation of the two sources (Figure 86b). Increases in transparency also improve the visual separation between the foreground and background. In our case, there are two depths: background and foreground. The background is at the greater perceived depth, based on contextual cues and occlusion cues, even though there is no real physical or visual depth present (the resulting combined image is simply blended).

We tested different degrees of transparency with a variety of backgrounds. We found that the subjective preference for degree of transparency depends on the information formats of the background and of the foreground. While the background information content is determined by the application task, 20%-50% transparency appears to work well in most application contexts. Within that range, both background and foreground objects are visible (divided attention is supported), and yet the contrast difference in background and foreground objects seems large enough to enable the user to separate out the two layers and focus on one (usually the foreground) when needed, while preserving an awareness of the other (background). This range was determined by asking users to test and evaluate a variety of different transparency levels and identify the point at which the background was no longer visible to them. We additionally asked users to evaluate different background content to identify the transparency level at which the foreground content became difficult to read. The suggested levels are in agreement with our formal Stroop experimental studies and with our Text Menu and Icon Palette experiments, which showed threshold transparency values of 20% and 50%. In fact, our prototype evaluations suggest that slightly *more* opacity is desirable (i.e., 25-30% to 50%), which we believe relates directly to the visual interference problems arising from more complex images or stimuli. Figure 88 shows a few examples of the different levels of transparency in this prototype system.

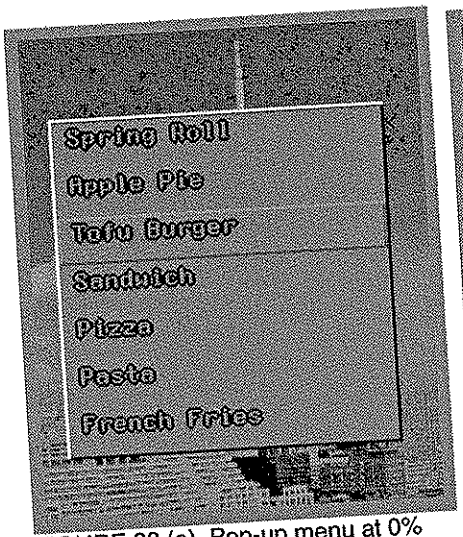


FIGURE 88 (a). Pop-up menu at 0%

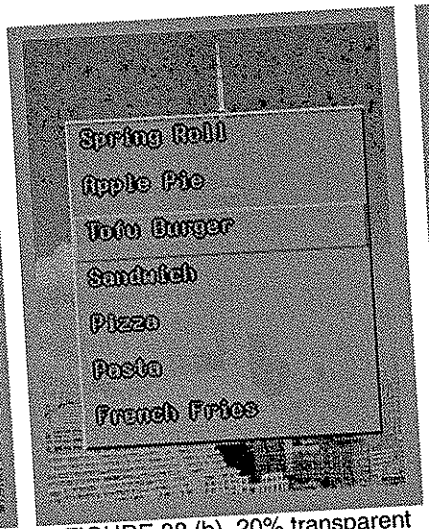


FIGURE 88 (b). 20% transparent

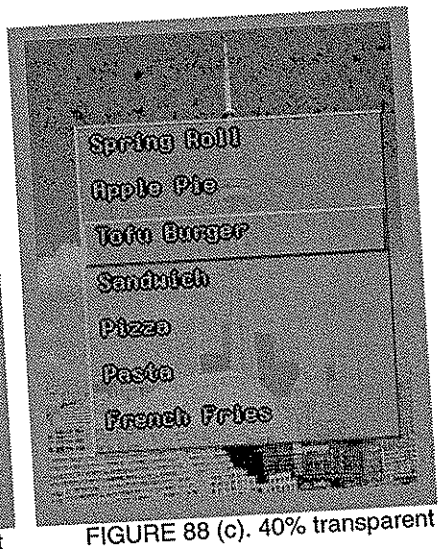


FIGURE 88 (c). 40% transparent

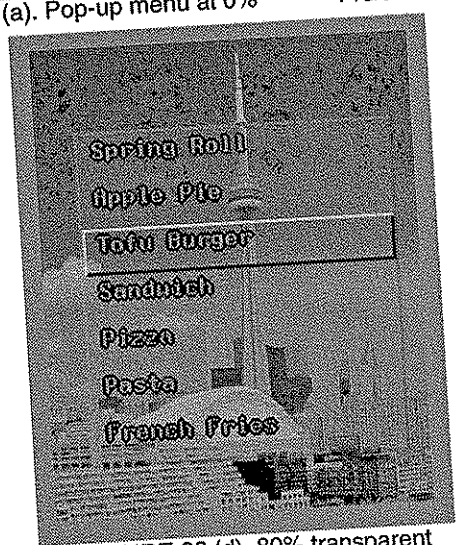


FIGURE 88 (d). 80% transparent

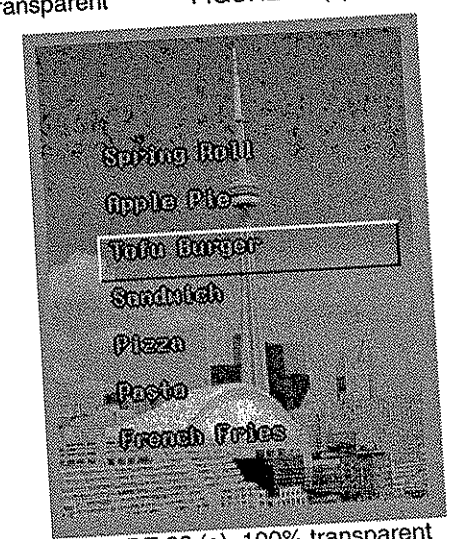
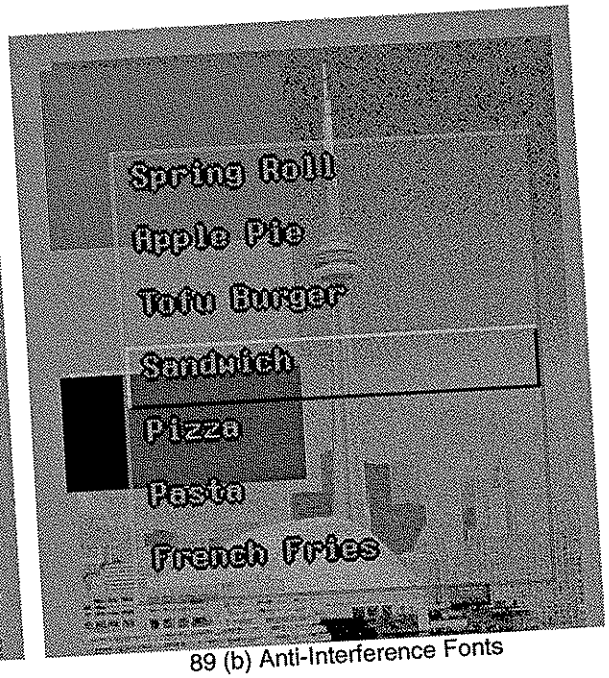


FIGURE 88 (e). 100% transparent

The usable levels of transparency seem to vary depending upon what content is contained in the background and how we draw the foreground menus (i.e., with or without outlines). (Usable in this context means that the text contained in the menus is legible.) In particular, when we generate a worst case scenario, either matched text over text or 100% transparent, black text menus over black images, we can predict poor legibility or no possible legibility at all (e.g., Figure 989a). This early insight into font design and text menus actually preceeded the Text Menu experiment, which formally confirmed the perception. Typically, we cannot predict the content of the background, and often, we cannot predict the color which might be aligned with any particular window or menu. Introducing an outline font has an immediate and remarkable effect on foreground legibility in virtually every condition or combination, particularly in these worst case scenarios (Figure 89b). However, such a change does result in slightly denser letters which potentially increases difficulty for background legibility.

FIGURE 89 (a) Regular Font



89 (b) Anti-Interference Fonts

The outline font we developed, called an "anti-interference" (AI) font, is shown in Figure 89b. It is well known that contrast is the most critical dimension of visibility. In AI fonts, the opposing outlines of the text are rendered in a color which has the maximal contrast to the color of the text. For any selected text color vector [R, G, B], our AI font algorithm calculates the luminance value Y according to the YIQ color model used in television broadcasting (Foley et al., 1990, p. 589). Note that the red, green and blue components are not equally weighted in contributing to luminance.

Based on the value of Y, our algorithm then determines the outline color with the maximal luminance contrast. In practice, only two color vectors can be the candidates for the solution: [0,0,0] (black) when $Y>Y_{max}/2$ or $[R_{max}, G_{max}, B_{max}]$ when $Y<Y_{max}/2$, where Y_{max} is the maximum luminance value and $R_{max}, G_{max}, B_{max}$ are the maximum red, green and blue values respectively.

Since an AI font has two opposing color components, it remains visible on any color background. Figure 89 (a,b) shows a worst case scenario when the background color is the same as the text or the text outline. As we can see, the AI font is "impenetrable" by any background color. However, one drawback of this new font is that each individual letter (or the stroke size) is virtually doubled.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.528 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Using this algorithm, we have also tested the width of the outline surrounding the text. A one pixel wide outline often does not seem to stand out from the background. Two pixel wide outlines show clearly on any background but introduce a loss in the sharpness or clarity of each character in the word. This could be corrected by changing the method of creating the outline, or by running any one of a number of sharpen functions on the resulting outline graphic. We implemented the outlines by simply offsetting one or two pixels and drawing a new pixel in the specified contrasting color for every letter. This is clearly not computationally efficient, though the user does not perceive any delays. There are numerous better implementations, though most require more programming effort. For example, we could have designed a special outline font using the GL utilities or X resources toolkits.

From the above informal evaluation, it was apparent that font styles and, in particular, outlining techniques can often compensate in poor visibility conditions when high levels of transparency are desirable. This was empirically confirmed in the Text Menu experiment which was based on these earlier developed hypotheses.

Finally, informal evaluation also indicated that practice with transparency improved performance. In particular, our more experienced menu users selected items based on spatial location and did not require high legibility. Also, users who were highly familiar with the background content seem to have less difficulty separating information into the correct layers. This expertise was acquired through using the prototype system over several hours.

6.4 Summary

Although fairly simplistic, our static and interactive prototypes gave us a number of insights into the design of transparent windows, icon palettes, and text menus. There were at least three major benefits derived from taking this approach.

First, we were able to quickly create environments that allowed us to informally evaluate variably transparent user interfaces and to identify sources of potential problems in the design. From this, we generated ideas about possible solutions and in some cases, we were even able to get informal feedback on the possible solutions.

Second, we formulated a new series of questions which helped us to define the next set of experiments and the hypotheses. While maintaining the progression from the original Stroop experiments in measuring attention and interference, these experiments related directly to the feedback we received from prototype evaluation (portions of which were completed prior to, or in parallel with, the experiments). This was as a result of using more complex and realistic

background and foreground content with variable transparency. We formulated a series of hypotheses related to content type (solid rendered objects, wire frame objects, or text objects), usable transparency levels, and text legibility. These experiments are described in detail in Chapter 5.

Third, we were able to partially assess the complexity of implementing transparency. This was important for two reasons. We needed to implement a whole new set of complex, variably transparent images for our subsequent experimental research, unlike the stimuli used in the Stroop experiments. These images required more sophisticated equipment (and hence a different operating system) from the Stroop experiments. The interactive prototype gave us a fairly accurate estimate of the effort to implement these systems, and a good idea of how to build them, without the added complexity of the new image libraries. We also wanted to implement transparency into an existing application for a realistic case study evaluation (Chapter 7). In this latter situation, we would integrating our transparency code into a substantially larger and more complex body of code. Based on the methods used in creating the Stroop experiments, combined with the totally different methods used in creating the interactive prototype, we had some experience in two different implementation strategies. The case study results from this implementation are described in detail in Chapter 7.

7.0 Case Study[1]

7.1 Description

The last phase of this research involves integration of the implementation techniques explored in Chapter 6, discussed in more detail below (and in Appendix A), and the integration of the parameter values derived from the empirical evaluations into a working application environment. This phase of the research was conducted jointly with Alias|Wavefront using their StudioPaint™ application in 3-D modeling and painting. The parameter values from the controlled experiments, and the design alternatives and implementation techniques from the prototypes were programmed into existing company source code to modify this product. An evaluation of existing StudioPaint™ users was conducted using this modified version.

While the experiments described in Chapter 5 incorporated stimuli from real products, they still reflect a strong element of experimental control, necessary for systematic empirical evaluation. Applying this research to real products is a much more valid test of the utility of transparency and the technologies required to support it. In real task contexts, the users have goals which directly relate to the productivity and efficiency of their tools, in this case a 3D modeling and paint system. They cannot afford to be tolerant of any interface changes which hinder this productivity, and thus are quite vocal about such changes. However, they are equally vocal about changes to the system which may improve their work.

In order to understand some of the constraints in our final released design, it is necessary to outline several key technical details which limited the scope of our implementation. These limitations have in turn driven a list of new operating system level features, currently under investigation for implementation in the future. We first briefly describe these challenges (more details are provided in the Appendix A). We then describe our particular implementation and the evaluation results.

---

[1] Portions of this chapter were published as Harrison, B. L. and Vicente, K. J. (1996). A Case Study of Transparent User Interfaces in a Commercial 3-D Modeling and Paint Application. In the Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting, September 4-8, 1996. Pittsburgh, PA. Santa Monica, CA: Human Factors and Ergonomics Society.

7.2 Technical Details

When creating two merged windows, a number of problems arise, since both of these windows are interactive. Traditionally, under opaque X windows, only the top window is visible and therefore the operating system stops updating the underlying piece of the background. In the case of transparent windows however, it is necessary to continue refreshing the underlying window if anything about it changes, *regardless* of whether it is under another window or not. Since we can see through the top window, the whole purpose is to be able to notice changes as they happen in the underlying layer. This means changing the way window updates are managed by the operating system. This can be fairly complex. However, we determined another method of managing windows which automatically solves some of these window refresh problems.

High end graphics computers are structured such that the display architecture is formed in layers or planes. A given machine has a specific number of bit planes allocated to displaying images. Some of these are reserved for the operating system's minimal requirements. However, many machines have overlay planes or pop-up planes. These lie on top of the usual display space. Typically, many applications write pull-down menus into overlay planes. These planes are most commonly used for transient user interface objects which appear temporarily. Writing windows to the overlay plane has the advantage that any windows in the display space (the layer below the overlay plane) continue to be updated and refreshed. This immediately simplifies the code for managing windows. However, there is a drawback - several, in fact. At present, it is not possible to alpha blend images into the overlay plane, so a stipple mask must be used (with the associated limitations described earlier about using stippling). Also, the number of colors and the availability of overlay planes is machine-dependent. It becomes necessary to determine what type of machine is running the transparency program in order to adjust the level of transparency and to determine whether to even attempt the adjustment. Finally, any window in the overlay plane has an associated color map which defines the colors that are needed by that window. While it is possible (and even typical) to have multiple windows with multiple color maps, every time the cursor moves over a new window, its color map is swapped in. This sometimes creates strange color artifacts in other windows and flashing on window borders and title bars. The best solution is to assign one color map to all objects in the overlay plane to ensure consistency.

Despite the aforementioned limitations, we used the overlay plane implementation with a stipple mask since other options limited performance. Note that in order for transparent interfaces to become prevalent, it will be necessary for standard overlay planes to be features of future computer designs across all hardware platforms.

7.3 Evaluation

Our evaluation was conducted in two stages. Approximately one month after the first release of the transparency version of StudioPaint™, we interviewed eleven users. Several modifications to the transparency algorithms and code were made based on their comments. We then waited an additional three weeks and again interviewed these same users.

A semi-structured interview was used. All users were asked the same questions about their usage of StudioPaint™to obtain a "User profile". (The specific questions are listed in Appendix C.) They were also all asked to list the three aspects of the interface they liked the most and disliked the most. Finally, they were asked for their opinions about transparency in particular. All responses were recorded. Those aspects of the interview relating to transparency are summarized below.

We intermittently video taped work sessions from randomly selected users at random times, both before and after the transparent user interfaces were introduced. These video tapes were analyzed to augment our interview data. We looked for continuing work patterns and changes in work patterns. The results of the video tape analysis are integrated into the results reported below.

7.3.1 User Profiles

The case study was conducted with product users who worked for the company that developed StudioPaint™ i.e., Alias|Wavefront Inc. We collected user profile information to categorize both levels of expertise and how extensively users had worked with the new transparent interface. Users were grouped into three categories, based on these profiles (Table 10). These categories were largely an artifact of working with a user community in the actual StudioPaint™ development company.

The first category was comprised of four users who worked primarily as programmers or product testers. Their use of StudioPaint™ was characterized by short "bursty" paint sessions, used to produce scribble style drawings. A large combination of features and painting tools was used, with each being used only a few times. Their task goal was to ensure that the system worked at a basic level, and that there were no apparent software bugs.

The second user group consisted of three "demo gurus". These users were primarily marketing and sales staff who were expert StudioPaint™ users, completely versed in the product's capabilities. Their task goal is to produce fast, interesting, and flashy demonstrations to potential

customers at trade shows. Few have artistic training. As a result, these users typically import almost completed, high quality artwork and demonstrate key features through fast, dynamic touch ups to the image. (A professional artist produces the images). The demonstration usage typically consists of short sessions which go through a few select features and any new capabilities.

The last user category is the group comprised of four hired artists who use StudioPaint™ to produce images. These users most accurately reflect the task goals and usage patterns of the StudioPaint™ customers. For this reason, we weighted their comments most heavily in our assessment. (Note that generally *both* novices and experts should be included in any analysis, or at minimum the most stringent criteria generated by either user group should be applied. In our application domain, expert users generated the most restrictive and demanding usage constraints and thus emphasis on their input was appropriate.) The artists' task goal is to produce high quality images. They accomplish this through lengthy sessions which span days, weeks and sometimes months, depending upon the image complexity. The process is similar to that outlined in the original scenario, presented in Chapter 1. They use some paint tools frequently, some occasionally, and some rarely or not at all. They heavily customize the system and the defaults to suit their particular task and work style. These defaults are often image based (e.g., wide brush strokes and specific color tones might be saved away). They are all highly skilled user of the StudioPaint™ system.

Type of User	# of Users	# who used transparency at beginning of trial	reason for NOT using transparency
programmers, testers	4	4	
product "demo" experts	3	1	2 users did not know how to turn feature on
professional artists	4	2	2 users turned feature off after one week of usage

Table 10. User profiles for subjects in case study.

In terms of the specific experience levels, this ranged from 6 months experience to more than 5 years of experience with StudioPaint™. Many of the users also have extensive experience on other high-end painting and drawing systems.

7.3.2. Results

Both prior to and after the introduction of transparency, the users maintained similar screen layouts. This consisted of a drawing area which was a full-screen window, with menu items and icon palettes minimized in size and pushed to either the top or the side of the drawing window (Figure 91). Interactive windows, overlapping the drawing area, were used to change tool attributes (Figure 92). According to the users, and in particular, the artists who were extensive expert users, "no screen size is big enough" and "anything which minimized the number and size of overlapping windows is good".

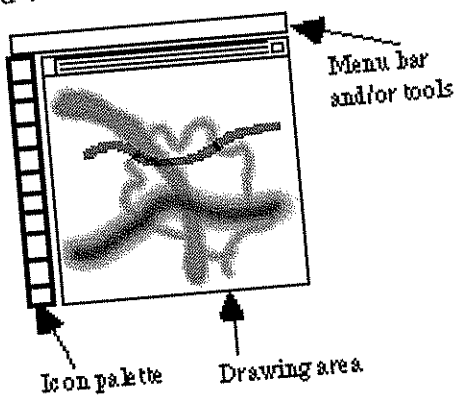


FIGURE 91. Diagram of typical screen configuration and layout of windows. Reflects task during drawing or painting, typical of a right handed user.

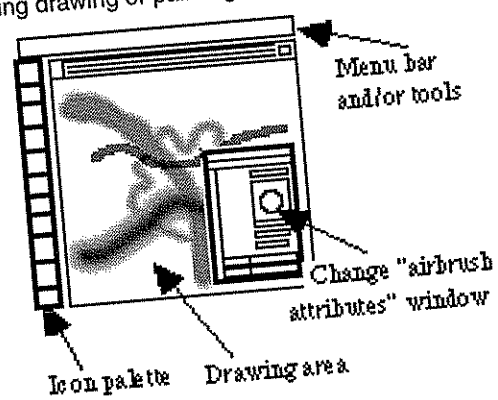


FIGURE 92. Diagram of typical screen configuration and layout with overlapping tool modifier window.

The first interview was carried out about 1 month after the transparency had been in use. It was discovered that about half of the users interviewed had not been using transparency (see Table 10 for summary). The people who did use it were: all four programmers/testers, one "demo guru", and two artists. The remainder of the demo users had not yet received information about how to turn the feature on. Two of the artists using transparency had turned it off after about 1 week of use. Specific comments and reasons are described below.

Issues related to technical limitations

There were some complaints that the font within the transparency windows was difficult to read, in particular when it was over dark images. This was attributed to using black fonts primarily combined with a dark gray or black stippled background color. The result was a distortion to the font edges. This was later fixed by changing the default background color to light gray, thereby avoiding conflicts with any of the font colors. Note that another solution to this problem would be the use of the Anti-Interference fonts used in both our prototypes and in our experiments described in Chapter 5.2. Unfortunately, it was not feasible to implement these new fonts within the time frame dictated by the scheduled product release dates. Therefore, we were limited to changes in font style or color, and to using fonts from the existing font library.

Another common complaint was about "flashing windows". Here, users were referring to an unfortunate artifact produced by drawing windows in the overlay plane. In particular, any window in the overlay plane can be assigned its own color map (a list of colors that the window uses). Whenever the cursor moves over a particular window, that window and its color map become active. If the color maps are different between windows, changes in the active window result in swapping in and out different color maps. This generates a flashing of the window borders and title bar, from highlighted to unhighlighted. Although it may seem a subtle change, to artists this was so annoying that one turned off transparency altogether. This problem can be partially addressed by assigning a common color map to any windows which use the overlay plane. However, some slight but noticeable highlighting changes will still occur which are meant to indicate the currently active window. The complete solution to this problem is unclear at this time.

Users requested the ability to set their own level of transparency by using something like a slider. Unfortunately, due to technical limitations this is not a feasible change. Using stippling, there are only five possible levels of transparency (since a 4-bit mask is used): all bits are on (opaque), all bits are off (clear), 25%, 50%, or 75%. The 25% and 75% levels introduce a grid artifact since all pixels in one row are either on (25%) or off (75%). These grid lines look terrible. The 100% level is visually quite confusing over complex imagery, since nothing separates the foreground window from the background drawing. Also, as our experimental results showed, fully transparent interfaces are not particularly good for foreground selection. This effectively leaves a 0% opaque and 50% semi-transparent level as the only usable and acceptable settings: in essence turn transparency off or turn it on to 50% (which was the way we implemented it). If we were able to use alpha-blending, we could have implemented a slider, since we would have had continuous levels of transparency and no grid artifacts.

Some users requested the ability to set their own tool window background color. We initially set the background color to a dark gray or black since this color most clearly showed the image underneath. However, this introduced font legibility problems with the black letters when combined with the stipple pattern. We changed it to light gray. We have not decided whether to provide this capability as yet. It is unclear if it would be wise to allow users to generate settings which accidentally, as a side effect, introduced problems with legibility. Of course, the users could change this setting. However, any setting affecting the window requires re-starting the StudioPaint™ system, making changes a lengthy process to invoke. This is technically feasible and still under consideration.

Users commented that some windows have very little transparency and many opaque objects still, limiting the utility of transparency. In the current design of the product, there are a number of objects within a window which we left opaque. These were objects which were selectable (e.g., buttons, sliders), objects which could be entered from the keyboard (e.g., text entry fields), and selectable areas, such as the color wheel. In particular, there are a number of embedded diagrams, such as the color wheel or the radius diagram for adjusting brush size, where precise selection and accurate visual perception are critical. In these cases, the objects were left opaque. The area around these objects was transparent. Windows which contain many such objects were primarily opaque as a result. We did implement a version of the system where the button and object borders were opaque, while the object itself was transparent. While this made substantially more of the window transparent, it was not obvious that the "buttons" and text fields were selectable to users. This was partly because users often equate "grayed-out" or partially invisible items with "un-selectable at this time". There are several other possible implementation solutions described in Chapter 8, which are unfortunately not technically possible at the moment.

A number of suggestions about dynamically changing the user interface mechanism were made. These are also covered in Chapter 8. Most of the suggestions made are technically impossible at this time.

Issues related to work flow

In general, transparency was felt to be most useful when windows were poorly placed or there were too many open. A typical session has anywhere from one to four windows open simultaneously (or a minimum of 10% of the screen obscured). One artist commented that, with experience, more than one open window should not happen often. Ideally, almost all the artists commented that *any* window was in their way. They wanted a system where no windows were

required so that they could continually work directly on their drawings with no obstructions. Transparency was seen to be one step in that general direction.

Most users commented that transparency works best in situations which are "static". By this they meant cases where they wanted visual feedback about some operation, as opposed to situations where they were actually trying to draw or paint directly behind the window. It still felt awkward to paint behind the windows in a continuous, uninterrupted brush stroke though nothing technically prevented this.

Several users wanted transparency in the "support" windows and dialogs. This would include the Help System windows and any message dialogs presented to the user. The message dialogs are generally system related issues. Serious errors would likely need to remain opaque to ensure that they were clearly seen. With such errors it is often important to interrupt the user's work. The company is currently working on making the Help system transparent.

7.4 Extensions and Implications

The exercise of implementing transparency in a product revealed more technical challenges than we had anticipated. We made a series of compromises in the design parameters to adjust for this. A number of operating system bugs were discovered and various temporary work-arounds had to be created until these bugs are fixed. Finally, we discovered that our implementation approach is greatly impacted by the hardware configurations of specific machines. In order to successfully implement transparent interfaces, we found it necessary to include procedures that determine the type of machine running the program and then adjust for it's capabilities (or lack thereof). With newer hardware, these procedures will likely need to be updated. However, we believe that this reasonably reflects the complexity of creating new technologies and integrating these into commercial products for trials. Many useful insights can be obtained and many creative solutions can emerge in such situations.

A number of the users' comments resulted directly from technical problems. It is clear that even seemingly minor issues to a programmer are highly irritating to someone with an artist's eye. These can influence the user to the point of shutting off the features! It is interesting to note that with transparency, it is sometimes difficult to "hide" the technical and implementation aspects from the user (as good UI practice might suggest). Again, this provided us with a number of valuable insights into our user group, and we were able to address many of the apparently aesthetic, yet critical, issues.

The case study confirmed our earlier experimental findings that default settings have an enormous impact on the usability of transparency. Window background color, font color, type style, and transparency level, interact and make the difference between a system that works and one that does not. Relatively minor adjustments result in major perceptual differences. We determined that 50% transparent windows in lighter colors seemed to work best (consistent with our earlier experimental results).

We did observe several new behaviors in the transparency case. Users often made drawing strokes which started outside an interface window and continued through the window. They did not attempt such strokes when using opaque windows. Additionally, we observed less window movement and fewer open/close operations with transparent windows. Transparent windows were moved or closed only when the user wished to initiate a drawing stroke directly under the window. Overall, transparent windows were more frequently moved rather than closed, whereas opaque windows were frequently closed to facilitate work.

We were not able to assess whether transparency has a significant impact on improving the user's overall productivity or work flow, as we had hoped. This is very difficult to measure. Since we implemented transparency into a new product release, it is very difficult to separate out which aspects of work are influenced by the introduction of transparency and which are influenced by some other changes to the system or its features. The main changes introduced (unrelated to transparency) were: new icon palettes and a complete reorganization of menu item ordering, hierarchic menus, user definable and save–able "toolkits" (collections of tools in a sequence), and user constructable and save–able pop-up menus.

There are mixed reviews about transparent windows. While most users seem to like them, there are still a number of technical issues which are problematic. This case study illustrates the challenges that one encounters in moving from basic research to commercial applications. We, and our test users, believe that transparent interfaces hold great promise, but it is equally clear that, for now, technical implementation details stand in the way of realizing those benefits. Therefore, further technical innovation is required before we can put the insights gained from our previous experiments into practice.

One major advantage of transparency is its application to other more sophisticated user interface tools. Transparent interfaces are a necessary first step to facilitate integration of more complex user interfaces which require transparent capabilities. They open the door of opportunity for entirely new functionality and novel tool sets such as click-through, see-through tool palettes and specialized see-through filters. These advanced ideas remain largely unexplored and untested, yet represent potentially powerful user interaction techniques. It is likely in conjunction with these newer user interface tools, that transparency can provide the most significant benefit.

8.0 Conclusions

As summarized in Chapter 1, Figure 5, this dissertation has presented a number of empirical studies, several prototypes, and a case study with the goal of better understanding transparent user interfaces and suggesting design guidelines. We have moved from a presentation of theories in visual attention and the Stroop Effect (Chapter 3) to theoretically motivated experiments (Chapter 4) and gradually evolved into more complex images in more representative experiments (Chapter 5). After assessing the implementation complexity in a simple programming environment (Chapter 6), we modified and evaluated a working commercial product (Chapter 7). This progression from theoretical to applied research has provided a number of contributions.

8.1 Contributions

In this dissertation, we have described a systematic research program aimed at investigating transparent user interface tools within a visual attention framework. We have proposed and applied a new design space framework (Chapter 2) which allows us to position current and future technologies relative to one another, while suggesting new technologies which have yet to be developed. The axes in this framework (depth, transparency, Cartesian displacement) offer new possibilities in terms of how current systems might smoothly evolve along one or several of these dimensions. It has proved to be a useful mechanism for organizing a wide variety of complex technologies and designs.

In starting from theories in focused and divided attention, we have defined a concept of fluency of work. This fluency indicates how smoothly or seamlessly integrated the tools are with the task goals and work methods. While the concept of seamlessness has been discussed by Ishii (Ishii, Kobayashi and Grudin, 1993), we have proposed several ways of measuring or assessing fluency within the context of any system design, not just transparent interfaces. In particular, some suggested measures are: time and frequency of explicit user interface window management activity, tool selection and de–selection, window re–sizing and movement, frequency of tool exchanges, and overall time spent directly on the task as it relates to total working time. The work space layer measures include: extent of awareness of changes to the work area (or portions of it), ability to identify and/or select salient features from the work space, and amount of time spent actively working. Additionally, there are a number of subjective measures to rate how

aware and frustrated (or not) users are with the current tool selection method and what they estimate the time spent is for actively working (or not).

Using the extensive body of literature in visual attention, interference, and the Stroop Effect, we derived a set of hypotheses and experiments which were progressively more representative of a target task. Our Stroop experiments (Chapter 4) presented a novel contribution to the Stroop literature and identified the most constraining limitations on visual interference. We additionally demonstrated a use for the Stroop Effect which supported product design and applied research. Specific design results were obtained as threshold values for transparent interfaces. For identification and selection of items in the foreground, performance levels off at 50% and does not deteriorate further with additional increases in transparency. There is a rapid performance degradation between 5% and 20% transparency. Our results suggest that for background focused attention tasks, substantial performance gains occur within the first 20-25% transparency and then levels off from 20% to 100%. Levels of 5% or less do not seem usable.

Our subsequent experimental work addressed more specific design issues related to visual interference based on information content and transparency (Chapter 5). We suggested a generalizable method of evaluating foreground selection tasks. We created an experimental paradigm which measures background awareness using feature identification and selection. This methodology could be applied outside of transparent applications to assess other aspects of interface design. By combining both the foreground and background experiments, we were able to determine optimal transparency performance points and trade-offs between foreground and background tasks. There is no apparent performance penalty (in terms of response time and error rate) for moving from opaque menus or icon palettes to 50% transparent. Text and wire frame icons or backgrounds perform similarly and both are much worse than solid images, which appear to be more interference resistant. The optimal point for text and line art icons placed over text and wire frame backgrounds is at approximately 60% transparency Solid icons have an optimal performance trade-off point with any background type at about 75% transparency. Regular Motif style menus have an optimal trade-off with background awareness at 50% transparency, whereas our AI fonts work well up to 90% transparent independent of background type. These results suggest that higher levels of transparency can be used by designers if they know the application task is characterized by solid images or forms (such as drawing or painting, video, photos, animation).

In addition to the experimental work, we built several prototypes and modified a commercial product. This has led to a number of implementation–specific contributions. In particular, we have discovered and reported several serious operating system bugs and we have documented the

need for new features in upcoming future hardware releases which will facilitate not only transparency but also screen refreshes, re–draws, and rendering processes. These features are being implemented by Silicon Graphics at the operating system level in both new hardware and new software. It seems likely that one day transparency will be a simple–to–apply feature not unlike choosing a window color.

In addition to the specific task domain we investigated, we believe that there are many other task domains, comprised of a primary work space and overlapping tools to manipulate that work, where transparent interfaces would be equally applicable. In fact, our analysis of existing system designs, presented in Chapter 3, reflects not only a lack of empirical data and design guidelines, but also indicates a wide variety of novel applications which make use of transparent layers. We believe that this thesis presents immediately applicable results, as well as stimulation for future research and evaluation.

8.2 Future Work

From the start, we had a number of ideas about dynamically changing context–sensitive interfaces. These changes could be based upon contextual cues, cursor position, or level of expertise. However, we discovered that implementing such changes is not technically feasible without several modifications to the operating system of the hardware configuration. We have summarized a number of intriguing possibilities here, though none have been evaluated.

8.2.1 Cursor–sensitive changes

Windows and selectable items need only be clearly visible at a time when it is appropriate to select something. If users move their cursor over a window and pause, this potentially signals an intent to perform an activity within that window. A transparent window could detect this status change and immediately increase its level of opacity. More specifically, as users move their cursor over a particular item, if that item is selectable or enter-able it could "highlight", or in the context of transparent interfaces, it could become opaque.

Clicking *through* a foreground layer, such as in the philosophy used in the ToolGlass work (Bier et al., 1993), is still possible with such proximity–sensitive cursors. When over a selectable object, a click would be interpreted to mean that the topmost function is selected. The system would then proceed to determine whether a meaningful object is aligned from the background layer and select that object as the target. (By analogy, this would be like shooting through all of the aligned positions in all of the layers in order). If the user clicks over a non-selectable region

of the foreground (e.g., surface area), the click is passed through to the background layer for interpretation.

The proximity of the cursor could also affect the level of transparency in a more continuous manner, instead of simply substituting the highlighting concept for an opaque–transparent transition. As the cursor gets further from the user interface objects these could "recede" or become progressively more transparent. Items that are closer are conversely more opaque. Items directly below the cursor would be completely opaque.

8.2.2 Task–sensitive changes

In addition to using cursor position, there are other contextual cues which could be used to influence whether users clearly see the foreground layer or the background layer. For example, if users are actively painting or drawing on a canvas, they are likely to not want windows blocking their brush strokes. As a brush nears a window which would obscure it, the stroke could "eat away" at the window. The area of the window which blocks the stroke would become highly or completely transparent. Presumably, this might work by taking some form of radius around the tip of the brush stroke and make it a "window eraser". Users would, of course, need some easy mechanism for recalling the window back to its original state.

Other contextual cues are currently available based on the status of the system and/or commands being executed. If users have just instigated some change to their work and the system is executing a screen re-draw or refresh, they likely need to see the result of that change before moving on to the next action. In this case, users would not want windows to block their view until they receive visual feedback.

8.2.3 Expertise–sensitive changes

Novice users often have special system settings to facilitate learning a new system, e.g., long menus versus abbreviated menu names, large icon sizes versus small icons, or expanded window views. With experience, users rely more on memory about the spatial relationships of items and their relative positions, and they do not need to read every item. To accommodate this learning, many systems have features which can be set for expert users. We could likewise have gradual increases in transparency levels, as expertise increased. We have already observed that many expert users merely need outlines or borders which provide very basic targeting information. Systems could either have user adjustable settings to accommodate this evolution, or the system could track the user's progress and dynamically make adjustments, or suggestions for adjustments, based on the number of hours logged.

8.3 In Summation

This dissertation provides an in-depth investigation of transparent user interfaces (Chapter 4 and 5). Experimental results suggest that, when properly designed, transparent tool palettes, menus, and windows can support accurate selection while still preserving a visual awareness of the underlying information. We put forward the notion that transparency is a useful interface mechanism which improves awareness of multiple layers of functionality, and that an understanding of user behavior with transparency is pre-requisite to designing more sophisticated see-through interface tools.

The intent of this thesis is to provide practical guidance for interface designers interesting in integrating transparency into their systems. To this end, we described factors affecting the choice of level of transparency, selection response time, and selection accuracy. We discussed the effect different information content has on visual interference and hence task performance. Through experimental work and prototyping, we illustrated methods for testing possible design combinations. Finally, we discussed some of the technical constraints and issues which must be taken into consideration in modifying a commercial application through one case study of such an implementation. This thesis has also provided an example of moving from basic empirical research to applied problem domains within the context of designing and evaluating transparent user interface tools. This combination of basic and applied research has made a novel contributions to the understanding of transparent interface technology, to the human factors issues pertinent to this technology, to the development of this technology, and to associated basic research issues.

9.0 References

Becklen, R. and Cervone, D. (1983) Selective looking and the noticing of unexpected events. *Memory and Cognition*, 11 (6), 601-608.

Berlyne, D. E. (1951). Attention to change. *British Journal of Psychology,* 42, 269-278.

Berlyne, D. E. (1960). *Conflict, arousal, and curiosity*. New York: McGraw-Hill.

Berlyne, D. E. (1970). Attention as a problem in behavior theory. In *Attention: Contemporary theory and analysis*. D. E. Mostofsky (Ed.), New York: Appleton-Century-Crofts, 25-49.

Bier, E. A., Stone, M. C., Fishkin, K., Buxton, W., and Baudel, T. (1994). A taxonomy of see-through tools *Proceedings of CHI'94*, Boston, Mass. April 24-28, 1994, New York: ACM Press, 358-364.

Bier, E. A., Stone, M. C., Pier, K., Buxton, W., and DeRose, T. D. (1993). Toolglass and magic lenses: The see-through interface. *Proceedings of SIGGRAPH'93*. Anaheim, CA. August 1-6, 1993. New York: ACM Press, 73-80.

Broadbent, D. E. (1957). A mechanical model for human attention and immediate memory. *Psychological Review,* 64, 205-215.

Broadbent, D. E. (1958). *Perception and Communication*. London: Pergamon Press Ltd.

Buxton, W. A. (1995). Integrating the periphery and context: A new model of telematics. In *Proceedings of Graphics Interface'95*, Quebec City, Quebec, May 17-19, 1995, Canadian Human-Computer Communications Society, 239-246.

Cohen J.D., MacWhinney B., Flatt M. & Provost J. (1993). PsyScope: A new graphic interactive environment for designing psychology experiments. *Behavioral Research Methods, Instruments & Computers,* 25(2), 257-271.

Dalrymple-Alford, E. C. (1972a) Associative facilitation and interference in the Stroop color-word task. *Perception and Psychophysics*, 11, 274-276.

Dalrymple-Alford, E. C. (1972b). Sound similarity and color word interference in the Stroop task. *Psychonomic Science*, 28, 209-210.

Desmond, J. (1987). HUD guidance improves landing performance. *Aerospace Engineering,* July, 1987, 48-53.

Deutsche, J. A. and Deutsche, D. (1963). Attention: Some theoretical considerations. *Psychological Review*, 70, 80-90.

Dyer, F. N. (1971). A comparison of chromatic and achromatic versions of the Stroop color-word test. *Psychonomic Science*, 22, 235-237.

Dyer, F. N. and Severance, L. J. (1972). Effects of irrelevant colors on reading of color names: A controlled replication of the "reversed Stroop" effect. *Psychonomic Science*, 28, 336-338.

Englebart, D. (1994). 1968 Conference, *CSCW'94 Conference Video Technical Program Proceedings*. October, 1994. Chapel Hill, NC. ACM SIGGRAPH Video Review number 108.

Eriksen, B. A. and Eriksen, C. W. (1974). Effects of noise letters on the identification of a target letter in a nonsearch task. *Perception and Psychophysics*, 16, 143-149.

Eriksen, C. W. and St. James, J. D. (1986). Visual attention within and around the field of focal attention : a zoom lens model. *Perception and Psychophysics*, 40, 225-240.

Eriksen, C. W. and Yeh, Y. (1985). Allocation of attention in the visual field. *Journal of Experimental Psychology: Human Perception and Performance*, 11, 583-597.

Fischer, E. (1979). The role of cognitive switching in head-up displays. *NASA Contractor Report 3137*. NASA Ames Research Center.

Foley, J.D., van Dam, A., Feiner, S. K., and Hughes, J. F., (1992), *Computer Graphics - Principles and Practice* (second edition), Reading, Massachusetts: Addison-Wesley.

Genau, A. and Kramer, A. (1995). Translucent history, Short paper in *Conference Companion Proceedings of CHI'95*. May 7-11, 1995. Denver, CO. New York: ACM Press, 250-251.

Gold, T. (1968). The head-up display as a correlate to the visual world. In M. Brooks (Chair) *Proceedings of the Symposium on Head-Up Displays: Human Factors Society Metropolitan Chapter*. New York: Riverside Research Institute, 1.1-1.38.

Greene, H. B. (1988). Manual CAT III with a HUD: Requirements and testing. *Proceedings of the 32nd Annual Symposium of the Society of Experimental Test Pilots*. Lancaster, CA, 223-238.

Gumenik, W. E. and Glass, R. (1970). Effects of reducing the readability of the words in the Stroop color-word test. *Psychonomic Science*, 20, 247-248.

Harrison, B. L., Ishii, H., Vicente, K. J., Buxton, B. (1994). Transparent layered interfaces: An evaluation of a display design space to enhance focused and divided attention. *Proceedings of CHI'95*, May 6-12, 1995. Denver, CO. New York: ACM Press, 317-324.

Harrison, B. L., Zhai, S., Vicente, K. J., Buxton, B. (1994). Semi-transparent user interface objects: Supporting focused and divided attention. *Cognitive Engineering Lab Technical Report CEL-94-08*, Dept. of Industrial Engineering, Univ. of Toronto, November, 1994.

Harrison, B. L. Kurtenbach, G., and Vicente, K. (1996). An experimental evaluation of transparent user interface tools and information content. *Proceedings of User Interface Software and Technologies UIST'96*, November 14-17, 1996, Pittsburgh, Penn. New York: ACM Press. 81-90.

Harrison, B. L. and Vicente, K. J. (1996). An experimental evaluation of transparent menu usage. *Proceedings of CHI'96*. May 6-12, 1995, Vancouver, B.C., New York: ACM Press. 391-398.

Harrison, B. L. and Vicente, K. J. (1996). A Case Study of Transparent User Interfaces in a Commercial 3-D Modeling and Paint Application. In *the Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting*, September 4-8, 1996. Pittsburgh, PA. Santa Monica, CA: Human Factors and Ergonomics Society.

Hockey, G. R. J. (1986). Changes in operator efficiency as a function of environmental stress, fatigue, and circadian rhythms. In K. R. Boff, L. Kaufman, and J. P. Thomas (Eds.), *Handbook of perception and human performance*. New York: John Wiley and Sons, 44.1-44.49.

Hughes, T. C., Hassoun, J. A. and Barnaba, H. M. (1992). A comparison of head-up and head-down display formats during instrument flying tasks. *ASC-TR-93-5003*. Wright-Patterson Air Force Base, OH: Crew Station Evaluation Facility.

Ishii, H. (1990). TeamWorkStation: Towards a seamless shared workspace. *Proceedings of CSCW'90*, Oct. 7-10, 1990, New York: ACM Press, 13-26.

Ishii, H. and Kobayashi, M. (1992). Clearboard: A seamless medium for shared drawing and conversation with eye contact. *Proceedings of CHI'92,* Monterey, CA, New York: ACM Press. 525-532.

Ishii, H., Kobayashi, M., and Grudin, J. (1993). Integration of interpersonal space and shared workspace: Clearboard design and experiments, *ACM Transactions on Information Systems,* Vol. 11, No. 4, October, 1993. 349-375.

Jensen, A. R. & Rohwer, W. D. (1966). The Stroop color-word test: A review, *Acta Psychologica,* 25, 36-93.

Johnson, J., Roberts, T. L., Verplank, W., Smith, D. C., Irby, C.H., Beard, M., and Mackey, K. (1989). The Xerox Star: A retrospective. *IEEE Computer* 22(9). 11-29.

Kabbash, P. and Buxton, W. A. S. (1995). The Prince technique. *Proceedings of CHI'95*, May 6-12, 1995. Denver, CO. New York: ACM Press, 273-279.

Kabbash, P., Buxton, W. A. S., and Sellen, A. (1994). Two-handed input in a compound task. *Proceedings of CHI'94,* Boston, Mass. April 24-28, 1994, New York: ACM Press, 417-423.

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

Kahneman, D. and Chajczyk, D. (1983) Tests of the automaticity of reading: Dilution of Stroop effects by color-irrelevant stimuli. *Journal of Experimental Psychology: Human Perception and Performance*, 9 (4), August, 1993, 497-509.

Kamlet, A. S. and Egeth, H. E. (1969). Note on construction of Stroop-type stimuli. *Perceptual and Motor Skills*. 29, 914.

Keele, S. W. (1972) Attention demands of memory retrieval. *Journal of Experimental Psychology*, 93, 245-248.

Klein, G. S. (1964). Semantic power measured through the interference of words with color-naming. *American Journal of Psychology*. 77, 576-588.

Knowlton, K. C. (1977). Computer displays optically superimposed on input devices. *Bell System Technical Journal*, Vol. 56, No. 3, March, 1977. 367-383.

Kobayashi, M. and Ishii, H. (1994). DispLayers: Multi-Layer display technique to enhance selective looking of overlaid images. Poster from *CHI'94 Conference*, April, 1994, Boston, Mass. April 24-28, 1994, New York: ACM Press

Kohler, P. A. (1972) *Aspects of Motion Perception*.. New York: Pergamon.

Kohler, W. (1929). *Gestalt Psychology.* New York: Liveright.

Kramer, A. (1994). Translucent patches: Dissolving windows. *Proceedings of UIST'94.* November 2-4, 1994. Marina del Ray, CA. New York: ACM Press, 121-130.

Kramer, A. F. and Jacobsen, A. (1991). Perceptual organization and focused attention: The role of objects and proximity in visual processing. *Perception and Psychophysics*, 50 (3).

Kramer, A. F., Wickens, C. D., and Donchin, E. (1985). Processing of stimulus properties: Evidence for dual-task integrality. *Journal of Experimental Psychology: Human Perception and Performance.* 11, 393-408.

Larish, I. and Wickens, C. D. (1991). Divided attention with superimposed and separated imagery: Implications for head-up displays. *Technical Report ARL-91-4/NASA HUD-91-1*, University of Illinois at Urbana-Champaign, Institute of Aviation.

Lauber, J. K., Bray, R. S., Harrison, R. L., Hemmingway, J. C., and Scott, B. C. (1982). An operational evaluation of head-up displays for civil transport operations. *NASA Technical Paper 1815.* Moffett Field, CA: NASA Ames Research Center.

Leiberman, H. (1994). Powers of ten thousand: A translucent zooming technique. Demonstration and short paper. In *Proceedings of UIST'94.* November 2-4, 1994. Marina del Ray, CA. New York: ACM Press, 15-16.

Lu, I. M. and Mantei, M. M. (1993). Managing design ideas iwth a shared drawing tool. In *Interacting with Computers: The Interdisciplinary Journal of Human-Computer Interaction.* Vol. 5, No. 1, March, 1993, 79-114.

MacLeod, C. M. (1991). Half a century of research on the Stroop Effect: An integrative review. *Psychological Bulletin,* Vol. 109, No. 2, 163-203.

MacLeod, C. M. and Dunbar, K. (1988). Training and Stroop-like interference: Evidence for a continuum of automaticity. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 14, 126-135.

Martin-Emerson, R. and Wickens, C. D. (1992). The vertical visual field and implications for the head-up display. *Proceedings of the Human Factors Society 36th Annual Meeting*, 1408-1412.

McCain, L. (1983). Color priming effect with concurrent tasks. *Bulletin of the Psychonomic Society*, 17, 8-9.

McCann, R. S., Foyle, D. C., and Johnston, J. (1993). Attentional limitations with head-up displays. *Proceedings of the 7th International Symposium on Aviation Psychology.* Columbus, Ohio: Ohio State University.

Moray, N. (1967). Where is capacity limited? A survey and a model. *Acts Psychologica*, 27, 84-92.

Moray, N. (1969) *Attention: Selective processes in vision and hearing.* New York: Academic Press.

Naish, J. M. (1963). Properties of a head-up display system relevant to approach and landing. *International Air Transport Association, IATA 15th Technical Conference*, Lucerne, Switzerland.

Naish, J. M. (1964). Combination of information in superimposed visual fields. *Nature*, 202, 641-646.

Naish, J. M. (1979). A review of some head-up display formats. *NASA Technical Paper 1499*. Moffett Field, CA: NASA Ames Research Center.

Naish, J. M. and Miller, D. L. (1980). An experimental evaluation of head-up display formats. *NASA Technical Paper 1550*. Moffett Field, CA: NASA Ames Research Center.

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

Neisser, U. and Becklen, R. (1975) Selective Looking: Attending to Visually Specified Events. *Cognitive Psychology*, 7, 480-494.

Newman, R. L. (1987a). Responses to Roscoe, The trouble with HUDs and HMDs. *Human Factors Society Bulletin*, 30, 3-5.

Newman, R. L. (1987b). Improvement of head-up display standards: Volume III: An evaluation of head-up display safety. *AFWAL-TR-3055*. Wright-Patterson Sir Force Base, OH: Flight Dynamics Laboratory.

Posner, M. I. and Snyder, C. R. R. (1975) Facilitation and inhibition in the processing of signals. In P. M. A. Rabbitt and S. Dornic (Eds.), *Attention and Performance V*. New York: Academic Press.

Regan, J. E. (1978). Involuntary automatic processing in color-naming tasks. *Perception and Psychophysics*, 24, 130-136.

Regan, J. E. (1979). *Automatic processing - Doctoral Dissertation*. As cited in MacLeod, 1991.

Roscoe, S. N. (1987). The trouble with HUDs and HMDs. *Human Factors Society Bulletin.*, 29.

Sanders, A. F. (1970). Some aspects of the selective process in the functional visual field. *Ergonomics*, 13, 101-117.

Sanders, A. F. and Houtmans, M. J. M. (1985). Perceptual processing modes in the functional visual field. *Acta Psychologica*, 58, 251-261.

Schmandt, C. (1983). Spatial input/display correspondence in a stereoscopic computer graphic workstation. *Computer Graphics*, Vol. 17, No. 3., 253-259.

Schneiderman, B. (1982). The future of interactive systems and the emergence of direct manipulation. Behavior and Information Technology. Vol. 1 (1982). 237-256.

Sichel, J. L. and Chandler, K. A. (1969). The color-word interference test: The effects of varied color-word combinations upon verbal response latency. *Journal of Psychology*. 72, 219-231.

Smith, D. C., Irby, C. H., Kimball, R., Verplank, W., and Harslem, E. (1982). Designing the Star user interface. *Byte* 7(4), April 1982, 242-282.

Steenblik, J. W. (1989). Alaska Airlines' HGS. *Airline Pilot*, December, 1989, 10-14.

Stone, M. C., Fishkin, K., and Bier, E. A. (1994). The Movable Filter as a User Interface Tool. *Proceedings of CHI'94*, Boston, Mass. April 24-28, 1994, New York: ACM Press, 306-312.

Stout, C. L. and Naish, J. M. (1967). The total systems concept for category III operations. *Proceedings of the 11th Annual Meeting of the Society of Experimental Test Pilots*. Los Angeles, CA. 79-105.

Stroop, J. R. (1935). Factors affecting speed in serial verbal reactions. *Journal of Experimental Psychology*, 18, 643-662.

Sutherland, I. E. (1963) SketchPad: A man-made graphical communication system. *AFIPS Conference Proceedings* 23, 323-328.

Tang, J. C. and Minneman, S. L. (1990). VideoDraw: A video interface for collaborative drawing. *Proceedings of CHI'90*, Seattle, Washington, April 1-5, 1990, NY: ACM Press. 313-320.

Thacker, C. , McCreight, E. M., Lampson, B. W., Sproull, R. F., and Boggs, D. R. (1979). Alto: A personal computer. In Siewiorek, Bell, and Newell, *Computer Structures: Principles and Examples*, Second Edition, McGraw-Hill, 549-572.

Thacker (1986). Personal distributed computing: The Alto and Ethernet hardware. *Proceedings of the Conference on the History of Personal Workstations*, New York: ACM Press. 87-100.

Treisman, A. M. (1969). Strategies and models of selective attention. *Psychological Review*, 76, 282-299.

Warren, R. E. (1972). Stimulus encoding and memory. *Journal of Experimental Psychology*, 94, 90-100.

Warren, R. E. (1974). Association, directionality, and stimulus encoding. *Journal of Experimental Psychology*, 102, 151-158.

Weintraub, D. J. (1987). HUDs, HMDs, and common sense: Polishing virtual images. *Human Factors Society Bulletin*, 30, 1-3.

Weintraub, D. J. and Ensing, M. (1992). Human Factors Issues in Head-Up Display Designs: The Book of HUD. *State of the Art Report, Crew System Ergonomics Information Analysis Center, Wright-Patterson Air Force Base*, Dayton, Ohio.

Weintraub, D. J., Haines, R. F. and Randle, R. J. (1984). The utility of head-up displays: Eye-focus vs. decision times. *Proceedings of the 28th Annual Meeting of the Human Factors Society*. Santa Monica, CA: Human Factors Society. 529-533.

Weintraub, D. J., Haines, R. F. and Randle, R. J. (1985). Head-up display (HUD) utility, II: Runway to HUD transitions monitoring eye focus and decision times. *Proceedings of the 29th Annual Meeting of the Human Factors Society*. Santa Monica, CA: Human Factors Society. 615-619.

Weitzman, B. (1963). A threshold difference produced by a figure-ground dichotomy. *Journal of Experimental Psychology*, 66, 201-205.

Wickens, C. D. (1984). *In Varieties of Attention*. Parasuraman, R. and Davies, D. R. (Eds). New York: Academic Press.

Wickens, C. D. (1991). *Engineering Psychology and Human Performance*. (Second Edition) Harper Collines Publishers.

Wickens, C. D. and André, A. D. (1990). Proximity compatibility and information display: Effects of color, space, and objectness on information integration. *Human Factors*, 1990, 32(1), 61-77.

Wickens, C. D., Kramer, A., Anderson, G. J., Glasser, A. and Sarno, K. (1989). Three-dimensional displays in focused and divided attention. T*echnical Report No. ARL 89-10.* University of Illinois at Urbana-Champaign, Institute of Aviation.

Wickens, C. D., Martin-Emerson, R. and Larish, I. (1993). Attentional tunneling and the head-up display. *Proceedings of the 7th International Symposium on Aviation Psychology.* Columbus, Ohio: Ohio State University.

White, L. C. (1969). Interference in identifying attributes and attribute names. *Perception and Psychophysics*, 6, 166-168.

Zhai, S., Buxton, W., and Milgram, P. (1994). The "silk cursor": Investigating transparency for 3D target acquisition. *Proceedings of CHI'94,* Boston, Mass. April 24-28, 1994, New York: ACM Press, 459-464.

Appendices

A. Implementation Strategies

In this section of the Appendix we provide some additional technical details which resulted from our prototyping efforts and from modifying the working commercial product. We discuss the implementation implications in terms of both complexity and performance.

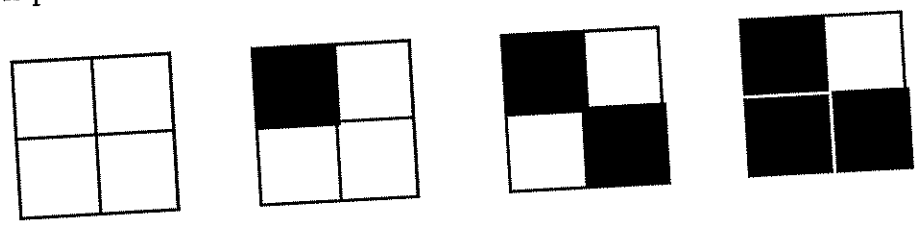
In our implementation, we use simple models of transparency that do not require realistic illumination effects such as refraction or diffusion of light. The application of more complex models which include lighting properties are largely inappropriate in this context, given that the fundamental purpose is to support awareness of multiple overlapping GUI objects. We do not need to specify lighting sources, light intensity and angles, surface substances and refractive properties, and the properties of nearby or underlying surfaces. This also reduces potential problems with maintaining image resolution, visibility, and legibility of underlying objects. By applying such simplifications, we obtain a more computationally efficient algorithm without compromising the intended purpose or functionality. For our purposes, we assume nonrefractive transparent surfaces with a distortionless view of underlying items.

Achieving nonrefractive transparency can be accomplished in a number of ways. Each method has strengths and limitations. Knowing the required extent of transparency in advance allows system designers to optimize on their choice of algorithm to maximize system performance efficiency. We will describe two preferred methods here, stippling using a bit-mask (a discrete algorithm) and alpha blending(a pseudo continuous algorithm).

A.1 Stippling or Dithering

For our purposes, we call any algorithm capable of producing only specific, discontinuous transparency levels, discrete. Most of these algorithms create a transparency effect by turning off and on various pixels thereby creating a "mask". These methods have been called dithering, stippling, XORing, and screen door transparency. Mask bits which are "on" choose a pixel from the foreground image; mask bits which are "off" choose a background image pixel. Masks always work using a binary decision criteria. The ratio of on/off pixels determines how much each of the foreground and the background image contribute to the resulting combined image. For example, as shown in Figure A1, in a 4-bit mask, 4 discrete levels are possible (and no other

levels in between). Using a mask where more bits can be specified will increase the number of discrete levels possible.



100% foreground | 75% foreground 25% background | 50% foreground 50% background | 25% foreground 75% background

FIGURE A1. A 4-bit stipple pattern mask.

Discrete transparency algorithms are simple to specify, a mask can be pre-defined and saved. The individual properties of either image, such as, color components, are not needed. Minimal computation is required since the mask merely directs the system where to take resulting pixel from (and the pixel content itself is unchanged). However, only a small number of levels (approsimately 10) are possible and the resulting combined image often has visually disruptive features such as grid lines and aliasing effects can appear. These artifacts result from the alignment of pixels taken from a single image plane. For example, applying a 4-bit mask across an image (4X2 mask applications) results in alternating grid lines of the background image (Figure A2). For this reason, the best results are achieved using a either a 50% semi-transparent mask which alternates on/off pixels or a dithered mask which is explicitly created to avoid these patterns.

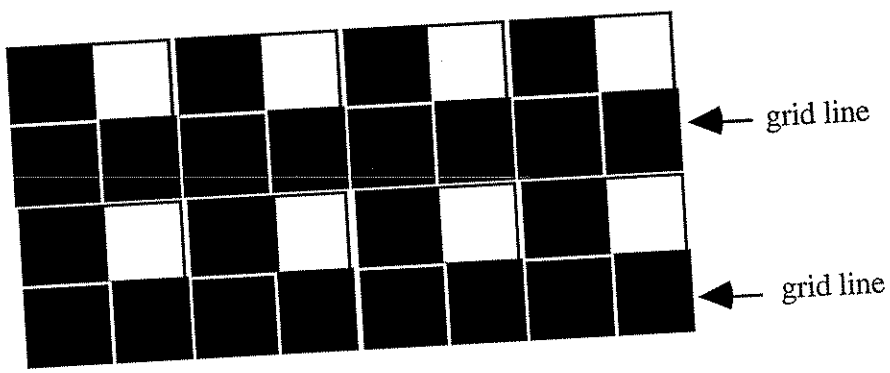


FIGURE A2. Applying a 4-bit stipple pattern at 75% transparent

Dithering was originally developed for printers which used to have (and sometimes still do have) only two intensity levels, yet must output complex gray scale images and photographs in a reasonable manner. Dithering can also be used to compute images displayed on CRTs. The idea is to simulate multiple levels of intensity by using the spatial integration performed by our optical

system. When the print or texture is fine and you view it from a slight distance your eyes tend to see a pattern and not the individual dots that make up the pattern. The fine detail gets averaged out into an overall intensity within regions in the image. In halftone or clustered dot dithering, the area (size) of each dot is determined in proportion to the intensity required. The size of these dots is generated by using patterns like those applied in stippling (Figure A1). There are a number of algorithms for computing dot area based on intensity values and subsequently spreading them in pseudo random positions to avoid aliasing, such that the resulting image appears reasonable. Dispersed-dot dithering is an extension of cluster-dot dithering in that the patterns used are not necessarily clustered. Instead specific matrices or randomized patterns are used (similar to applying 16-bit stipple masks) to avoid grid lines or unwanted texture artifacts. As in the stippling case, dithering is still a discrete method of masking out bits.

Our experimental work determined a number of transparent optimum points, based on the information content of the foreground and background layers. By applying these optimal transparency threshold values, determined in advance of implementation time, we can predict the size of the bit-mask required and the configuration for this mask (i.e., how many bits are on and off within the mask). For example, to achieve a 50% optimal level we can use an alternating on/off 4-bit mask. To achieve a 90% transparency effect, we require a 16-bit mask (or better). However, the larger the mask, the lower the resolution of the two GUI objects that can be seen.

A.2 Alpha Blending

For our purposes, we call an algorithm capable of producing any transparency level, continuous. Most of these algorithms create a transparency effect by computationally combining or blending attributes of the foreground pixel with attributes of the background pixel in a pre-determined proportion. The resulting new pixel is the combined image. These methods have been called interpolated transparency, alpha blending, or filtered transparency. Any transparency level may be specified and no graphical artifacts result; usually 256 levels is sufficient to provide a seemingly continuous scale, hence our use of the term pseudo continuous. For this reason, alpha blending algorithms are more flexible than discrete algorithms. However, they require more computational resources are therefore less efficient in terms of system performance.

In general, the continuous class of algorithms require some form of off-screen rendering or buffering and rendering. For this reason, even with hardware to support increased rendering speeds, these algorithms are much slower than the discrete algorithms.

In our static prototype implementation and for the images we used in our experiments, we applied an alpha blending algorithm to compute resulting pixels based on the combined R, G, B,

α values for both the foreground and background image pixels. Every pixel has a red, green, blue and alpha component. The formula for alpha blending is:

$$I = \alpha\ I_1 + (1-\alpha)I_2$$

where I is the resulting color intensity, I_1 is the color intensity of the foreground image pixel, I_2 is the color intensity of the background image pixel, and α is the specified transparency level between 0 and 1 (0= clear and 1 = opaque). Any value between 0 and 1 may be specified in practice, rounded off to 1/256. The resulting blended image appears completely smooth and free of unwanted aliasing effects and grid line artifacts. Again, using the pre-determined optimal transparency levels, it would be possible to implement the specific algorithm with a pre-determined α value in machine code or hardware to accelerate the computation of resulting blended images.

A.3 Using Overlay Planes

As we described earlier, SGI workstation are architected in such a way that application programmers have access to bit-planes of varying depth. There are a number of such planes relevant to implementing transparency. All IRIS™ workstations have two pop-up planes, intended as buffers for transient drawing for simple objects like pop-up menus. The *pop-up planes* will always appear visually superimposed over whatever is drawn in standard X windows. All workstations also have either 0, 2, 4, or 8 -bit *overlay planes*. Again the 2-bit overlay plane is most often used for transient objects like pop-up menus (it also does not have any color properties available). 2-bit overlays can only draw up to 3 colors. Most UI widgets require at least 4 colors. Any machine with an 8-bit overlay can present more complex objects including color components. Any item written to the overlay plane automatically appears superimposed on top of other windows.

Any item using the overlay plane must be assigned a color map, either by default or explicitly by the programmer. Each window can have its own color map. There are graphics routines (GL) for manipulating and assigning color maps. As mentioned in the Case Study (Chapter 8), using different color maps for multiple objects in the overlay plane can result in incorrect colors being used to draw the UI objects, flashing title bars, and sometimes unusual screen refreshes with unexpected colors applied to the desktop or windows. Color map switching occurs when input focus switches among the windows with different color maps (input focus is determined by whatever the cursor is hovering over, selection is not required). The size of the color map, and hence the number of color that your object can be, is determined by the number of bit planes

(e.g., 2-bit, 8-bit). When overriding the default color map, you also need to be aware of any macros which assume you have a standard set of colors. It is possible for a pixel to attempt to access an out–of–range color causing an incorrect color to be displayed. Furthermore, this use of color maps assumes that any modifications exist within a single application. There is no general official way for multiple applications to cooperate to use common color maps or default color maps. According to SGI documentation, one problem with the X11 visual model is that it does not consider layering and therefore applications which do this are nonstandard. Furthermore, users are warned that it is possible that none of the interface builders will work properly when using non-default visuals (which overlay planes are). There are a number of known bugs in the SGI IM resource manager which can also occur when using non-default visuals. There is no standard way to open windows in overlay planes. Once overcoming the issues around color maps and the overlay plane and the number of bits you have available, the next challenge appears. The resources describing the visual properties of a window can only be set at window create time – they cannot be later altered.

Thus far we have been able to successfully program transparent windows into the overlay plane, given the technical limitations discussed above and those mentioned in Chapter 8. We test at application start-up for the type of machine and the hardware it has before deciding whether to turn on the transparency feature. If the machine has an 8-bit overlay plane (or more) transparency is enabled.

While the above qualifiers may seem discouraging the advantage of using overlay planes is large. With normal X windows, in order for changes in the underlying window to appear the application programmer needs to write code to handle a new set of conditions which pass a window damage condition along. Window damage events must happen whenever either the top window contents change *or* the underlying window contents change (normally this occurs only in the former case because windows are assumed to be opaque). This damage event in turn must be interpret into a request for a window re–draw. In essence the number of events and re-draws is doubled hence the large performance penalty. One solution is to execute these re-draws is a very fast hardware buffer however, those do not exist on most SGI machines at this time. It would call for off-screen rendering which is only available on the very high end machines (e.g., Reality Engines).

B. Instructions for the Stroop Experiments

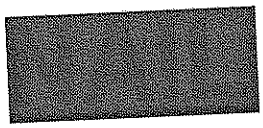
B.1 Color Naming Instructions

Opening instruction screen at the start of experiment:

In this experiment you will see a series of color patches in either red, blue, green, or yellow.

The color patches will be either blank (color only) or will contain a familiar word.

Sometimes the ink that the word appears in will be very "bright" and clear, other times the word will seem faded out. For example,



Name the **COLOR– ignore the word.**
Your verbal responses will be timed using the microphone provided.
Please try to do this as ACCURATELY and quickly as possible.

If you make an error press the RED button.

You do not need to correct your errors.

You will have a chance now to try a few sample trials...

<press the GREEN button when ready...>

Following practice trials, a "pause" screen appears:

If you have any questions at this point, please ask them.
****you will not be able to ask them while the trials are running ****

The experiment will last approximately 45 minutes. You will have rest breaks after every 15 minutes. The trials in the experiment will be similar to the ones that you just did.

If you are ready to start the experiment - click the GREEN button.

End of experiment, final screen:

Please remember to sign the video consent form before you leave.

Thank-you for your participation in this experiment.

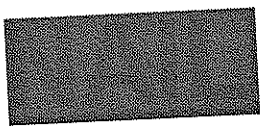
Your help is greatly appreciated!

B.2 Word Naming Instructions

Opening instruction screen at the start of experiment:

You will see a series of words. Most appear on red, blue, green, or yellow color patches. Some words appear with NO color patch.

Sometimes the ink that the word appears in will be very "bright" and clear, other times the word will seem faded out. For example,



Name the **WORD -- ignore the color. There is ALWAYS a word present. If you cannot see any word say "NONE" and press the YELLOW button. Please try to do this as ACCURATELY as possible.**

If you make an error press the RED button.

You do not need to correct your errors.

You will have a chance now to try a few sample trials...

<press the GREEN button when ready...>

Following practice trials, a "pause" screen appears:

If you have any questions at this point, please ask them -
****you will not be able to ask them while the trials are running ****

The experiment will last approximately 45 minutes. You will have rest breaks after every 15 minutes. The trials in the experiment will be similar to the ones that you just did.

If you are ready to start the experiment - click the GREEN button.

End of experiment, final screen:

Please remember to sign the video consent form before you leave.

Thank-you for your participation in this experiment.

Your help is greatly appreciated!

C. Questionnaires

This section includes sample questionnaires and subject consent forms which are representative templates of those used for each of the 6 experiments. In each case the description was changed to reflect the purpose and method of that particular experiment. The sample templates and subject consent forms were pre-approved by the Human Ethics Committee at the University of Toronto prior to running any subjects. We include only one sample of the questionnaires for the 6 experiments here and an outline of the questions used in the Case Study for the semi-structured interview.

Sample Experimental Consent Form
Department of Industrial Engineering
University of Toronto

Director of Experiment:	Beverly Harrison, M.A.Sc.	Dept of Industrial Engineering
Supervisor of Research:	Professor Bill Buxton	Dept. of Computer Science
	Professor Kim Vicente	Dept of Industrial Engineering

Purpose of Experiment:

The experiment you are about to participate in is set up to try to understand how people work new kinds of display technology. In the experiment you will be asked to perform two simple tasks, speaking into a microphone. In one situation, you will be asked to name the word which appears on the computer. In the other situation, you will be asked to name the color displayed. Sometimes this task will be more difficult than others because the word will appear faded out. We anticipate that the experiment will take approximately 45 minutes of your time.

Your participation in this experiment is entirely voluntary and you may ask questions or ask for clarification at several indicated points during the experiment. You may withdraw from the experiment at any time without loss of pay should you wish to.

We would like to videotape the computer screen over your shoulder. You will not appear on the video tape although your voice will be recorded. Both the video and audio tapes will serve as the record of this study and will facilitate our analysis of the experiment. All of these recordings and any questionnaires will be kept completely confidential and no written publication or presentation of this work will identify you. The experimental records will be viewed by the researchers only.

We appreciate your willingness to be a part of this study. In exchange for your voluntary participation you will receive $___ for your time. If you have any questions regarding the study and your participation in it, an experimenter will be available to answer them. If you agree to participate in this study and allow us to collect the information we have indicated, please sign and date this form below. Thank you again for your help.

Consent:

I understand the tasks I will be asked to perform in the experiment and that the information collected on me will be held in complete confidence. I agree to participate in this experiment and to have this information collected.

Date:_____

Name: (please print)_____

Signature: _____

Witness: (please print):_____

Witness Signature: _____

Videotape Presentation Consent Form
Department of Industrial Engineering
University of Toronto

Director of Experiment:	Beverly Harrison, M.A.Sc.	Dept of Industrial Engineering
Supervisor of Research:	Professor Bill Buxton	Dept. of Computer Science
	Professor Kim Vicente	Dept of Industrial Engineering

In most of our experiments an important part of the data we collect are video recordings of the participants interacting. As part of this experiment we have taken a videotape record of your interactions with the computer. At this point, we assure you that the videotape record will be held in complete confidence and will only be observed by the researchers involved in the experiment.

However, sometimes it is useful for us to present video data at research conferences or as part of publications in order to demonstrate a research point or to demonstrate the analysis process we used. To ensure confidentiality, the identities of the persons in our videotapes are withheld and we do not identify anyone on the videotape by name. We would like the opportunity to use segments from the video tape we have just taken of you for this purpose, should they provide illustrative examples.

Please check off the boxes below which authorize the exact usages which you are willing to allow us. Checking the boxes and signing this form indicates that you agree to allow us to present brief segments of video data for the purposes described. (Boxes left blank will be taken to mean no permission is given for that purpose.) This is often an important part of our work and has proved very valuable to us in the past, for other experiments.

If you wish, you may first have the opportunity to preview the videotape that was just made, and, if this is acceptable to you, sign this consent form allowing us to use the video tape for such presentations. Please indicate that you would like to preview the tape by checking the box.

- ❑ **I wish to preview the videotape**

I agree to allow presentation of the video tapes (where my identity is withheld) for the following purposes:

- ❑ **presentation to colleagues, internal presentations**
- ❑ **inclusion with the final dissertation work (of which this is a part)**
- ❑ **presentation at research conferences**
- ❑ **presentations at small research seminars (<30 people)**
- ❑ **distribution to other research labs (on request) as part of sharing similar research findings**
- ❑ **distribution as part of a video tape publication series (where the tapes may be sold to other researchers, universities, or laboratories)**
- ❑ **public relations or fund raising for our research laboratory**

I have read and understand the above description and I consent to permit the experimental video tape of me to be shown for research purposes as described above.

Date:_____

Name: (please print)_____

Signature: _____

Witness: (please print):_____

Witness Signature: _____

Sample Experiment Information Sheet
Department of Industrial Engineering
University of Toronto

Director of Experiment:	Beverly Harrison, M.A.Sc.	Dept of Industrial Engineering
Supervisor of Research:	Professor Bill Buxton	Dept. of Computer Science
	Professor Kim Vicente	Dept of Industrial Engineering

Description of Experiment:

Thank you for participating in our research. In this experiment you were asked to (1) name a word while ignoring the color, and (2) name the color while ignoring the word. In the first case, we are interested in the point at which transparent displays make words illegible. In the second case, you may have noticed some words were harder to ignore than others, for example, the word RED when you needed to name the color BLUE. This is called the Stroop Effect. We are interested in the point at which the opacity of the display helps you to name the color – even when the more difficult interfering words appear. The results from this experiment are being used to help us to design a new type of computer system which uses see-through windows and menus.

If you would like more detailed information on this experiment, or if you have any questions, please feel free to ask.

Thank you for your help with our study. You have contributed considerably to our research.

Acknowledgment of Receipt of Payment

I acknowledge that I have received the sum of $___ for participation in this experiment.

Date:_____

Name: (please print)_____

Signature: _____

Witness: (please print):_____

Witness Signature: _____

Questions for the Semi-Structured Case Study Interview

How would you characterize your role as a user of StudioPaint™?
(e.g., programmer, tester, demo person, artist, novice user, ...)

_____

How long have you used StudioPaint™?

_____ years _____ months _____ days _____ never

How long is your typical session with MagicPaint?

____ less than an hour ___ 1to 4 hours ____ 1 day ____ 2 to 4 days _____ more

How many of the features of the system do you think you use often or on a regular basis?

_____ 20-40% _____ 40-60% _____ 61%-80% ____ > 80%

How would you characterize your use of MagicPaint? (short fast sessions, long drawn out sessions over days, sporadic, rare,...)

_____

_____

How many UI windows do you typically have open during a session?

____ 1 or 2 ____ 3 or 4 ____5 or more

Have you used transparency?

If no, why not? _____

If yes, for how long? _____

If you answered NO to the above question, please go to the end of the questionnaire.

Did you continue to work in the same way or in a different way?

_____

If different, how?

_____

Has your screen layout remained the same or changed? _____

If changed, how?

_____

_____

Have your UI window sizes become larger, remained the same, or become smaller?

_____

Do you use find you use more/less/the same screen real estate for UI windows?

_____

_____

When do you find you rely on the transparency most, if at all?

_____

What happens when you want to make a brush stroke through an area where there is a UI window? (stop & move window, draw through window, pre-plan motion path, close window...)

_____

What do you like best about the new system?

_____

What do you like least about the new system?

_____

Are there things you cannot currently do that you wish you could?

_____

Do you have any other comments related to transparency in general or the way the MagicPaint system uses transparency?

_____

_____

D. Statistics Terminology

When comparing statistical means, typically we want to know if mean A is significantly different from mean B. When we perform one such comparison test (often called a comparison of means) we typically use a t-test (or Student t-test). Usually, the test determines whether the difference between the two means is equal to 0 (i.e., they are equivalent) or not. When we wish to compare multiple means, in theory we could use multiple t-tests but combining the results increases the likelihood of error. Also it is difficult to derive a single numerical value based on the difference between multiple variables. Comparing multiple means is typically done with an Analysis of Variance (ANOVA). Rather than test the difference between means, the F-test compares the variances of each treatment condition. Both the t-test and the F-test assume normally distributed independent population for each treatment condition and random selection from within the treatment population.

The **F value** is the ratio test which represents: (treatment effect for variable of interest + individual differences + experimental error) divided by (individual differences + experiment error). Conceptually it contrasts the variability *between* treatments against the variability in data *within* a particular treatment condition. The variance between treatments is assumed to be caused by three things: the treatment effect, individual differences, or experimental error. The variability within a treatment measures the variability expected by change (i.e., that caused by individual differences and/or experimental error). The F-test is a ratio of these two amounts. Mathematically it is computed using the mean of a sum of squares, which is the summation squared for an entire set of N scores or measurements. These sums are averaged across the entire set of measurements to give the mean value used in the F-ratio. The precise mathematical derivation is given below:

$$SS = \Sigma X^2 - \frac{(\Sigma X)^2}{N} \qquad \text{mean square MS} = \frac{SS}{df}$$

$$\text{F value} = \frac{\text{MS between treatments}}{\text{MS within treatment}}$$

df - represents the degrees of freedom (the extent to which that treatment is free to vary). This is the amount that the numerator and the denominator in the statistical test are free to vary independently for a particular set of scores or measurements. For any given treatment with n samples, only n-1 of these samples can vary. The last sample is forced to a particular value in

order to maintain the correct mean value for the sample. In general, small degrees of freedom mean that the data distribution for the F-test is more spread out. This tends to weaken the statistical test since more values will fall outside the cut-off point. High degrees of freedom increases the power of the F-test.

Computationally df is defined as:

df-total = N -1 where N is the number of trials in the entire experiment

df-total breaks into two component parts - the df-between-treatments = k-1, and the df-within-treatments = N-k. The df-within-treatments further breaks down mathematically to: df-between-subjects = n-1 and df-error = (N-k) - (n-1).

We normally list the result of an F-test as F(<df between treatments -numerator>, <df within treatment - denominator>) = f, $p < \alpha$. From the above equation, a large numerator relative to the denominator implies the treatments varied by more than chance. The smaller the F value, the less likely a statistically significant effect has occurred. Determining the cut-off point for acceptable ranges of F values is done by choosing an alpha level – p.

p is defined to be the probability that the result given would occur solely by chance. As such, it is an indicator of the statistical confidence that the factor under measurement had an influence on the resulting performance measured and this influence is not merely due to error in measurement technique or the normal random fluctuations that occur in data. In our experiments, we used a very conservative alpha value of $p < .01$ and usually are results show $p < .001$ meaning the probability that we see an effect which was just due to change is less that 1% (or even 0.1%).

There is one final statistical test which we relied upon extensively as a post-hoc test. Post-hoc tests are done only on variables or interactions which have previously shown a statistically significant effect in the t-test or F-test. Most post-hoc tests (such as a Tukey test, Sheffé, or Student-Newman-Keuls) determine the minimum difference between two means that is necessary for statistical significance. This minimum number is then compared against the experimental treatment mean specified. As explained earlier, if the mean difference between these two numbers exceeds this minimum value there is a significant different in your treatment variable. Post-hoc tests allow for slightly more precise and conservative investigation to rule out chance variation causing an incorrect statistical effect. As in any other comparison test, an alpha level is also specified for post-hoc tests. We used a conservative SNK post-hoc test with an $\alpha = 0.05$.

Cognitive Engineering Laboratory

CEL TECHNICAL REPORT SERIES

CEL 93-01 "Egg-sucking, Mousetraps, and the Tower of Babel: Making Human Factors Guidance More Accessible to Designers"
• Kim J. Vicente, Catherine M. Burns, & William S. Pawlak

CEL 93-02 "Effects of Expertise on Reasoning Trajectories in an Abstraction Hierarchy: Fault Diagnosis in a Process Control System"
• Klaus Christoffersen, Alex Pereklita, & Kim J. Vicente

CEL 94-01 "Cognitive 'Dipsticks': Knowledge Elicitation Techniques for Cognitive Engineering Research"
• Klaus Christoffersen, Christopher N. Hunter, & Kim J. Vicente

CEL 94-02 "Muddling Through Wicked Problems: Exploring the Role of Human Factors Information in Design"
• Catherine M. Burns

CEL 94-03 "Cognitive Work Analysis for the DURESS II System"
• Kim J. Vicente & William S. Pawlak

CEL 94-04 "Inducing Effective Control Strategies Through Ecological Interface Design"
• William S. Pawlak

CEL 94-05 "Research on Factors Influencing Human Cognitive Behaviour (I)"
• Klaus Christoffersen, Christopher N. Hunter, & Kim J. Vicente

CEL 94-06 "Ecological Interfaces for Complex Industrial Plants"
• Nick Dinadis & Kim J. Vicente

CEL 94-07 "Evaluation of a Display Design Space: Transparent Layered User Interfaces"
• Beverly L. Harrison, Hiroshi Ishii, Kim J. Vicente, & Bill Buxton

CEL 94-08 "Designing and Evaluating Semi-Transparent 'Silk' User Interface Objects:
Supporting Focused and Divided Attention"
• Beverly L. Harrison, Shumin Zhai, Kim J. Vicente, & Bill Buxton

CEL 95-01 "An Ecological Theory of Expertise Effects in Memory Recall"
• Kim J. Vicente & JoAnne H. Wang

CEL 95-02 "Strategic Plan - 1995"
• Cognitive Engineering Laboratory

CEL 95-03 "Cognitive Engineering Laboratory Profile"
• Cognitive Engineering Laboratory

CEL 95-04 "A Field Study of Operator Cognitive Monitoring at Pickering Nuclear Generating Station-B"
• Kim J. Vicente & Catherine M. Burns

CEL 95-05 "An Empirical Investigation of the Effects of Training and Interface Design on the Control of Complex Systems"
• Christopher N. Hunter

CEL 95-06 "Applying Human Factors to the Design of Medical Equipment: Patient-Controlled Analgesia"
• Laura Lin, Racquel Isla, Karine Doniz, Heather Harkness, Kim J. Vicente, & D. John Doyle

CEL 95-07 "An Experimental Evaluation of Transparent Menu Usage"
• Beverly L. Harrison & Kim J. Vicente

CEL 95-08 "Research on Factors Influencing Human Cognitive Behaviour (II)"
• Christopher N. Hunter, Michael E. Janzen, & Kim J. Vicente

CEL 95-09 "To the Beat of a Different Drummer: The Role of Individual Differences in Ecological Interface Design"
• Dianne Howie

CEL 95-10 "Emergent Features and Temporal Information: Shall the Twain Ever Meet?"
• JoAnne H. Wang

CEL 95-11 "Physical and Functional Displays in Process Supervision and Control"
• Catherine M. Burns & Kim J. Vicente

CEL 96-01 "Shaping Expertise Through Ecological Interface Design: Strategies, Metacognition, and Individual Differences"
• Dianne E. Howie

CEL 96-02 "Skill, Participation, and Competence: Implications of Ecological Interface Design for Working Life"
• Peter Benda, Giuseppe Cioffi, & Kim J. Vicente

CEL 96-03 "Practical Problem Solving in a Design Microworld: An Exploratory Study'
• Klaus Christoffersen

CEL 96-04 "Review of Alarm Systems for Nuclear Power Plants"
• Kim J. Vicente

CEL 96-05 "DURESS II User's Manual: A Thermal-hydraulic Process Simulator for Research and Teaching"
• Lisa C. Orchanian, Thomas P. Smahel, Dianne E. Howie, & Kim J. Vicente

CEL 96-06 "Research on Factors Influencing Human Cognitive Behaviour (III)"
• Dianne E. Howie, Michael E. Janzen, & Kim J. Vicente

CEL 96-07 "Application of Ecological Interface Design to Aviation"
• Nick Dinadis & Kim J. Vicente

CEL 96-08 "Distributed Cognition Demands a Second Metaphor for Cognitive Science"
• Kim J. Vicente

CEL 96-09 "An Experimental Evaluation of Functional Displays in Process Supervision and Control"
• Catherine M. Burns and Kim J. Vicente

CEL 96-10 "The Design and Evaluation of Transparent User Interfaces: From Theory to Practice"
• Beverly L. Harrison

CEL 97-01 "Cognitive Functioning of Control Room Operators: Final Report"
• Kim J. Vicente, Randall J. Mumaw, & Emilie M. Roth