InVision: a System Using Vision Patterns to Understand User Attention

by

Michael W. Li

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degrees of

Bachelor of Science in Computer Science and Engineering

and Master of Engineering in Electrical Engineering and Computer Science

at the Massachusetts Institute of Technology

February 6, 2001

Author_____
Department of Electrical Engineering and Computer Science
February 6, 2001

Certified
by_____
Ted Selker
Thesis Supervisor

Accepted
by_____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

InVision: a System Using Vision Patterns to Understand User Attention
by
Michael W. Li

Submitted to the
Department of Electrical Engineering and Computer Science

February 6, 2001

In Partial Fulfillment of the Requirements for the Degree of
Bachelor of Science in Computer Science and Engineering
and Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

This research demonstrates a new approach that uses patterns of eye motion for object selection in an eye tracking interface. The analysis of eye motion on the pattern level can deliver three values to an eye tracking interface: selection speed, reliability, and a more complete understanding of user attention. Current eye tracking interfaces use fixation as a means of target acquisition and/or selection. There are several problems with this approach concerning issues of selection speed, system reliability and the understanding of user attention. This research builds a system, called InVision, to demonstrate how the analysis of eye fixation at the pattern level can help provide solutions to these problems. First, selection speed is quick through the use of pattern identification as a means of selection. Second, pattern correlation can add reliability to an eye tracking interface. Finally, the ability to understand the context of a user's eye motion is provided through pattern interpretation.

Thesis Supervisor: Ted Selker
Title: Associate Professor, MIT Media Laboratory

Table of Contents

List of Figures

1. Introduction

Analyzing patterns of eye motion can deliver three values to an eye tracking interface: selection speed, reliability, and a more complete understanding of user attention. Eye fixation is the primary means of identifying user attention in present day eye tracking. Fixation is also commonly used for control purposes such as selection in an interface. There are several problems presented by this eye fixation technique including selection speed, system reliability and the understanding of user attention. This research proposes the use of eye fixation patterns as solutions to these problems. Selection speed is increased through the use of pattern identification as a means of selection. Pattern correlation can help improve the reliability of an eye tracking interface. Finally, the ability to understand the context of a user's eye motion is provided through pattern interpretation.

For the purposes of this research, an eye tracking interface tool called InVision is built that uses an eye pattern analysis approach. Specifically, this interface uses patterns of eye fixations to analyze and interpret eye motion data. Such an analysis moves beyond simple eye fixation identification and examines how the eye has moved around an image in the context of that image. Using the InVision interface tool, the three values offered by patterns that are proposed by this research are investigated.

First, the performance of an eye pattern analysis approach in attentive selection is quantitatively evaluated. An interface called the Eye Selection Tester is built for this purpose using the InVision system. The performance of the pattern approach is experimentally compared to that of an approach using simple fixation for selection. Two

variables are measured: selection accuracy and selection time, reflecting system reliability and system speed respectively.

The second part of the research qualitatively studies how examining a user's eye fixation patterns can reveal a more complete understanding of user attention. The Kitchen InVision project studies patterns of fixations for the purposes of identifying, interpreting and responding to user cognitive state. By understanding user attention on a more complete and contextual level, systems will be capable of interaction on a much higher-level by being able to better predict and accommodate a user's interests, tasks and questions.

2. Background

In order to understand this research and appreciate the results of the InVision project, it is necessary to have some background knowledge of the human eye, eye motion as well as the area of eye tracking pertinent to the research. Relevant background information is provided for the purposes of both helping the reader understand the specific research being addressed and providing a context for the work. In addition, this section defines eye and eye tracking terminology that will be used in the paper. The following sections provide a brief summary of the human eye and eye tracking, focusing specifically on different aspects of eye movement and how eye motion relates to user cognitive state and attention.[1]

2.1 Fixation and Saccade

Eye motion generally is not smooth as one might think, and is instead made up of a sequence of sudden jumps called *saccades*, and followed by *fixation*, a period when the eyes are relatively stable. Fixations and saccades characterize attentive and voluntary eye motion.

A saccadic jump takes only a fraction of a second (between 30-120 milliseconds) and generally does not exceed a visual angle of 20° from the previous point of focus. Wider angles of eye rotation occur both when looking at moving objects and when the observer is close to the object of focus. Research indicates that the time intervals between separate jumps averages between 0.2 and 0.5 seconds but may be significantly longer as well depending on the visual angle traversed.

Eye fixations represent the periods when the eye is focused on a particular location. Even during fixation however, the eye is constantly moving. During a fixation, the eye seems to be steadily focusing on one point, while in reality it is making many imperceptible jumps, resulting from a type of eye movement (see Section 2.2 and 2.3). This jittery movement generally falls within a one-degree radius of the fixation. Fixation durations are generally between 200 and 600 ms, after which another saccade is initiated (Jacob 1995).

2.2 Types of Eye Movement

Generally, one is not aware of the intricate patterns of eye movement involved with perceiving a scene. The true complexity of the vision system is transparent to the observer. In order to use eye tracking as a channel to understand the user, one must first understand the characteristics of eye movements. There are 7 different types of eye motion that have been identified: convergence, rolling, saccades, pursuit motion, nystagmus, drift and microsaccades, and physiological nystagmus (Bruce & Green, 1990). These different eye movements are described in detail below.

Convergence: This motion generally results when an object being foveated changes distance from the observer. When an object is close to an observer, the eyes adjust to point towards each other. While this motion can be voluntary, it is generally the result of focusing on a moving stimulus.

Rolling: An involuntary motion allowing rotation along the axis passing through the fovea and pupil. This motion is often influenced by angle of the neck.

Saccades: The sudden, rapid eye motion responsible for moving the eyes to a different area of the visual scene. While saccades can be voluntarily initiated, once a saccade is started, its path and destination cannot be changed.

Pursuit motion: This motion acts to keep a moving object foveated and is a much slower and smoother motion than a saccade. Pursuit motion eye movement cannot

be initiated voluntarily and requires a moving object passing across the visual field.

Nystagmus: Eye pattern response to the turning of the head.

Drift and microsaccades: Occurs during fixations. Motion is comprised of small drifts and correcting microsaccades. Involuntary movement.

Physiological nystagmus: Occurs during the fixation period for the purpose of oscillating the eye to continuously shift the object image on the retina. This shift is physiologically necessary to see an object and is explained further in the next section (Section 2.3).

Eye tracking data is the result of the combination of all of these movements. While saccades and fixations result from processes of attention, the other motions are the result of physiological conditions. A good analysis of eye motion will filter the voluntary attention movement from the many other involuntary eye motions that are involved with the process of perception.

2.3 The Role of Eye Movement in Vision

Eye movement plays an important role in perceiving an object, whether the object is stationary or moving, near or far. Eye movement is achieved by six muscles that can turn the human eye about any axis passing through the center of the eyeball. The visual axis of both eyes is always directed to one object and thus both eyes move concomitantly. Eye tracking technology depends on the vision's necessity for eye movement. The movement of the retinal image is necessary for the reception of visual information for two reasons discussed below.

Eye movement has some responsibility on a very low-level for maintaining vision. In order to process a scene, the eye requires constant movement due to the physiological limitations of the eye. When the eyes appear to be looking at a single point, they are

actually making a series of abrupt jumps. Research has demonstrated that impulses along the optic nerve occur only at the moment when the image changes on the retina (Babsky, Khodorov, Kositsky, & Zubkov, 1975). During constant application of light on visual receptors, impulses quickly cease along the corresponding optic nerve fibers and vision effectively disappears. For this reason, eye motion contains incessant imperceptible jumps that constantly displace the retinal image. This series of jumps stimulate new visual receptors and produce new impulses on the optic nerve, ultimately enabling the process of vision.

Another role that eye motion plays in vision results from the structure of the human eye. The retina contains millions of light sensitive cells known as rods and cones. The eye has approximately 7 million cones and 110-125 million rods. The cones are concentrated at the center of the retina, and a depression within this area called the fovea consists almost entirely of cones. The fovea is an area of high acuity, which extends over a visual angle of approximately one-degree. This means that, humans can only observe something in great detail at the middle of their retinas. The human eye constantly saccades about a scene, moving this area of acuity around in order to collect details about a visual scene. The complexity of the object being looked at is directly related to the complexity of the scan path of the eye movements (Babsky et al., 1975).

2.4 Cognitive State and Eye Movement Relationship

Research categorizes eye movement patterns into four different types: spontaneous looking, task-relevant looking, orientation of thought looking (Kahneman, 1973) and

intentional manipulation looking. These four different patterns of motion are described below:

Spontaneous looking: Spontaneous looking is the perception of a scene without a specific task in mind. The gaze pattern during spontaneous looking is directed to the parts of the image with the most information.

Task relevant looking: This type of looking results from the observation of a scene with a particular task or question in mind. The eye-gaze pattern on a particular scene is directly correlated to question being asked. (see Figure 1).

Orientation of thought: When an observer is not paying attention to what the eyes are looking at and the eyes are moving instead in reaction to thoughts. For example when asked to spell "MOTHER" backwards, some subjects move their eyes from right to left as if reading the letters from right to left in the word "MOTHER" that was visualized in their head (Stern, 1993).

Intentional manipulation looking: This results from the movement of the eyes to a specific location or in a specific way, with the intention of manipulating something through the action of the eye motion (Engell-Nielsen, & Glenstrup, 1995).

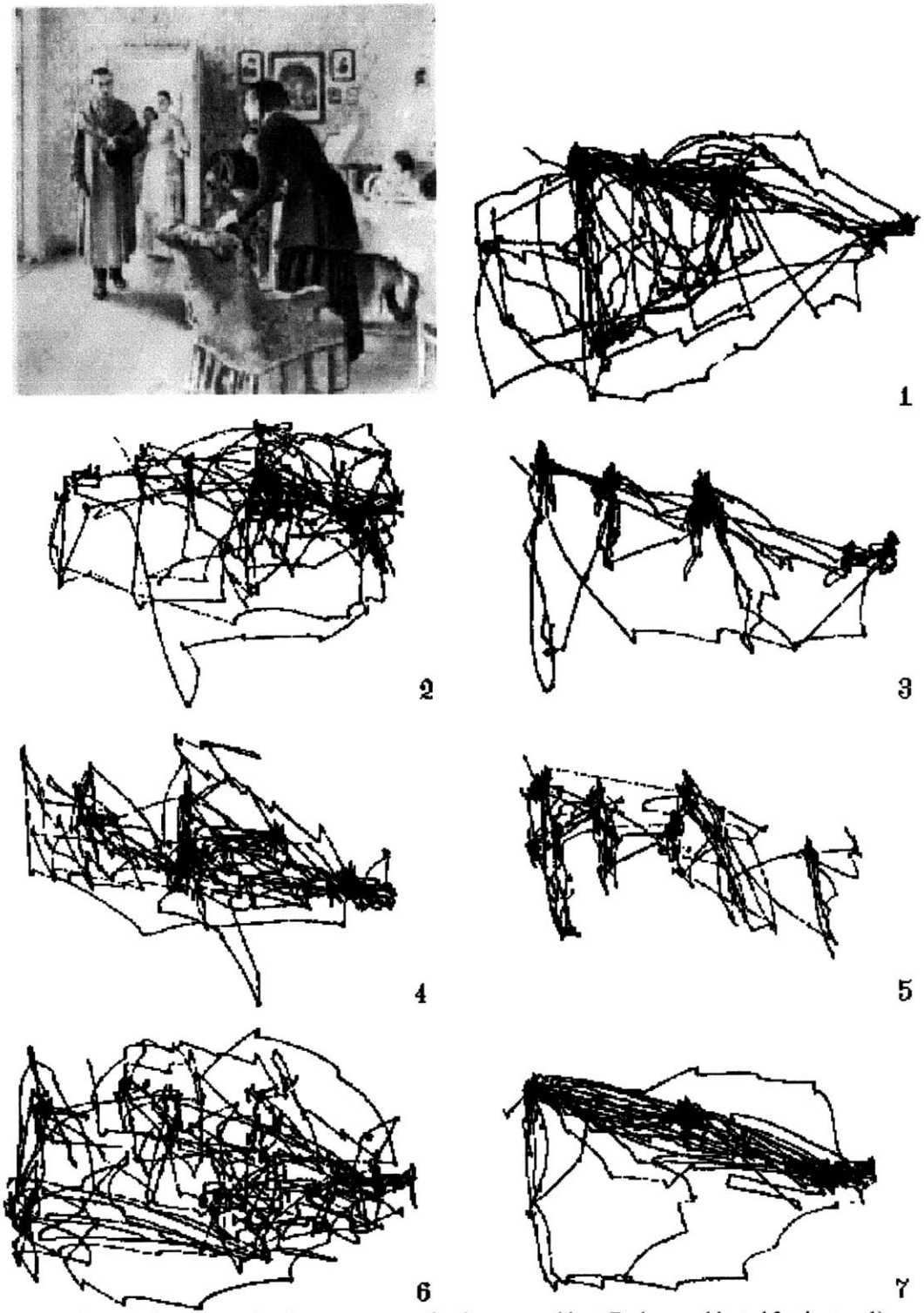

Figure 1: Seven records of eye movements by the same subject. Each record lasted 3 minutes. 1) Free examination. Before subsequent recordings, the subject was asked to: 2) estimate the material circumstances of the family; 3) give the ages of the people; 4) surmise what the family had been doing before the arrival of the "unexpected visitor;" 5) remember the clothes worn by the people; 6) remember the position of the people and -objects in the room; 7) estimate how long the "unexpected visitor" had been away from the family (Yarbus 1967).

People are generally interested in what they are looking at when in spontaneous and task relevant looking. When a person is performing some task that requires pulling information from the surrounding environment, the eye-gaze is generally correlated to what is being processed (see Figure 1). It follows naturally to question whether such a correlation holds in the opposite direction. What can a person's eye movement explain about cognitive state? Eye movements reflect thought process, and indeed a person's thought may be followed to some extent from eye movement analysis (Yarbus, 1967). In other words, eye tracking can provide a "window" into the cognitive thought processes of an individual.

How does the visual system decide where to fixate next? Two different stages within the attention selection mechanism have been identified: the pre-attentive stage and the attentive stage. The pre-attentive stage operates across the entire visual field whereas the attentive stage focuses only on one object, or at best a few objects, at a time. When objects pass from the pre-attentive stage to the attentive stage, they are "selected." During the fixation time, small initial saccadic adjustments are made, and it is thought that a decision is made during this interval as to where future fixations will be located (Barber, 1976). Attention is directed to a new location even before a saccade is initiated to move the eyes. Because of this, the movements of the eyes should not be considered as the selection process itself, but merely as the outcome of the attention selection processes preceding actual eye-shifts (Theeuwes, 1993). Fixations, while usually an indication of user attention, are not necessarily so, a topic discussed in a later section (see Section 3).

2.5 Fixation and Eye Motion Pattern in Eye Tracking

Fixation is an important phenomenon that is commonly used by eye tracking systems to provide an indication of local user attention. Most eye tracking interfaces use fixation as the basis for *target acquisition* which is the task of identifying a particular object. Target acquisition is generally an aspect of selection for eye tracking interfaces as well. Some eye tracking interfaces use eye fixation for both target acquisition as well as *selection*, such as IBM's Suitor project (Blue eyes: Suitor). The Suitor project, also known as Interest Tracker, distinguishes between a "glance" and a "gaze" by the amount of time a user has fixated on a particular area and uses a gaze as a means of selection. Techniques using fixation duration, or *dwell-time*, as a means of selection generally use a threshold value between 250-1000ms (Edwards, 1998). If a fixation lasts longer than the chosen threshold value, a selection is initiated. Interfaces using fixations for either for target acquisition or selection are referred to in this paper as *fixation-based* interfaces.

Fixation detection is an entire sub-field of eye tracking research. While the physiological concept of a fixation is understood, there are many different algorithms that exist for fixation detection (S. Zhai, personal communication, February 1, 2001). A technique for fixation detection is used in this research that is similar to the fixation recognition approach described by Jacob (1995). It is believed that the specific choice of the fixation algorithm will not affect the results and conclusions of the research presented.

More recently, research has been performed regarding the use of eye motion patterns to attempt to understand user cognitive state. One of the earliest uses of eye motion pattern to understand user attention is in a system created by Starker and Bolt

(1990) that displays a planet from "The Little Prince," a book by Antoine de Saint Exupery. It uses patterns of natural eye movement and fixation to make inferences about the scope of a user's attention. Edwards (1998) uses eye movement pattern identification in the Eye Interpretation Engine, a tool that can recognize types of eye movement behavior associated with a user's task. Perhaps one of the works that is most relevant to this research is done by Salvucci (1999) who describes a technique called fixation tracing, a process that infers user intent by mapping observed actions to the sequential predictions of a process model. This technique translates patterns of eye movement to the most likely sequence of intended fixations. In addition to the work presented in this paper, additional interfaces were implemented as preliminary research to this work. These interfaces studied grouping, patterns of association and identification in search. The use of eye motion patterns is still a relatively new research area that will become more prevalent as eye tracking technology improves.

3. Problems with Eye Tracking Interfaces

Fixation-based interfaces encounter three problems in the task of interpreting user eye movement:

1. Slow interface speed
2. Unreliability of is unreliable
3. Inadequate understanding of user attention

Each of these problems is discussed individually below.

3.1 The Problem of Interface Speed

The speed of an interface is also hurt by eye-gaze inaccuracy and imprecision. A system's ability to correlate recorded fixation with the actual point of fixation can greatly influence the system's response time. A system using dwell-time as a means of target selection (see Section 2.5) requires the user to wait for the recorded fixation to fall on the observed object before a system response is initiated. This amount of time can be quite large depending on the size of the object being selected and the feature mapping fixation algorithms being used. This poses a problem for fixation-based interfaces that uses only recorded fixation as a means of selection. Due to off-center fixations (see Figure 2), the selection time for a small object in such an interface might be large, or worse, the target object might never be selected. While there are algorithms that can recognize off-center fixations and map them to the intended targets, this gain in speed comes at a cost of accuracy (see Section 3.2).

Eye-gaze tends to be both too inaccurate and imprecise. A user's eye fixations, as recorded by an eye tracker, are often not centered over visual targets. There are two

reasons for this inaccuracy: users can fixate anywhere within a one-degree area of the target and still perceive the object with the fovea (see Section 2.3), and eye-trackers have a typically accuracy of approximately one-degree (Salvucci, 1999). Imprecision is introduced into eye-gaze data through the involuntary jittery motions produced by the eye (see Sections 2.1, 2.2, 2.3). Eye control is neither accurate nor precise enough for the level of control required to operate today's UI widgets such as scrollbars, buttons, hyperlinks and icons. Zhai, Morimoto, and Ihde (1999) show that the area represented by a user's fixation is approximately twice the size of a typical scrollbar, and much greater than the size of a character.

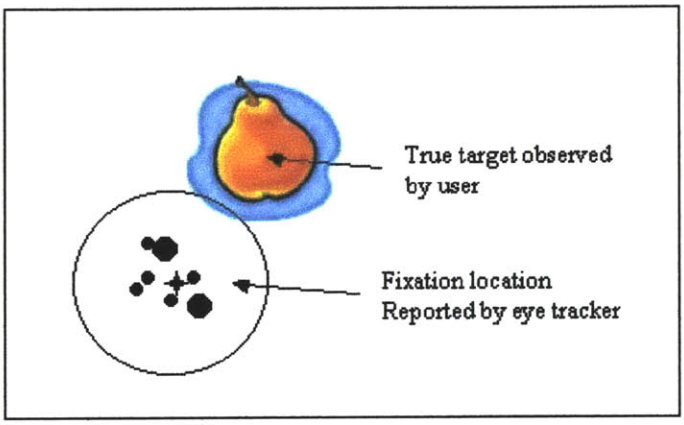

Figure 2. A sample of eye movement that shows the inaccuracy and imprecision related to eye-gaze.

Fixation-based interfaces are limited in their ability to handle and interpret eye tracking data because of this noise. By watching only for fixations, these interfaces adopt an easy technique to filter the noise, but at the same time end up ignoring important data as well. Consider the following problem: a user's eye gaze remains fixed on a point on the screen, but the eye tracking calibration is slightly off, resulting in a off-center fixation somewhere other than the location intended (see Figure 2). A fixation-based interface cannot effectively accommodate such noise.

3.2 Unreliability of Eye Control

Even with the best eye tracking technology, fixation-based eye control is very unreliable. This is due largely to the fact that the eye has evolved into a highly developed sensory device rather than something originally intended for control. The eye is a very good input device, but not a very good output device. While some of the eye movement involved in vision is voluntary, there is involuntary motion involved in vision as well (see Section 2). Eye movement must be interpreted carefully so as to avoid eliciting an unwanted response. For example, interfaces using a dwell-time method for selection, run the risk of being very unreliable. Users of such interfaces have to be aware of where and how long the eyes look at a particular object. If the eye fixation duration is not long enough, the target object will not be selected. Incidental fixations (Salvucci, 1999) that are not intended to actuate commands can select targets regardless of the user's intention. This is referred to as the "Midas touch" problem (Jacob, 1995), which is essentially the problem of distinguishing intended fixations from incidental fixations. Because of this limitation, users must be careful not to casually look at any target for too long or it will be selected.

Off-center fixations, described in the previous section (see Figure 2) can undermine eye control reliability as well. An off-center fixation could fall on an object other than the one intended, producing an incorrect system response. Algorithms and techniques exist that attempt to map off-center fixations to intended targets. Similarly, cluster analysis methods can help determine likely target areas of attention (Goldberg & Schryver, 1995). These techniques do not always produce correct results however, especially as the

number of target objects increase and the size of the target objects decrease. This again might result in the selection of an unintended target, building unreliability in the system.

Another related problem with such interfaces is that the user is often forced to change or adapt his/her eye motions to the interface. Human eye-gaze is not calm and controlled like the movement of a mouse. The fixation-based approach is not well suited for natural human eye motion that tends to be very rapid and unconstrained. Instead, the eyes saccade rapidly from point to point, making the task of keeping the eyes fixed on a specific point unnatural and perhaps straining.

3.3 Inadequate Understanding of User Attention

The final problem discussed regarding fixation-based eye tracking interfaces is an inadequacy for determining the location of user attention. A person's focus of attention has been traditionally thought of in the eye tracking realm as directly related to a person's eye-gaze direction. Gaze direction is typically determined by two factors: the orientation of the head and, the orientation of the eyes. One problem facing fixation-based eye tracking systems is deriving the user's focus of attention from the low-level eye-gaze patterns. While most research literature indicates that people are interested in where they are looking, this is not always the case. Due to the distribution of fovea receptors, it is not possible to tell where within approximately a one degree circle that a person is looking. In fact, a person can move his/her attention around within an area that is smaller than the fovea without making any eye movements at all (Jacob, 1995).

Even if a perfect eye tracking system existed, the problem still exists of how to find a user's focus of attention using only the gaze information. What a user is looking at and

what the user is mentally processing is not always related, and a perfect correlation between the two cannot be assumed. A high-level user model is needed to deal with involuntary eye movements and to understand eye movements on a larger scale. Scope of attention is another concept that can be better understood through a different analysis of data than can be offered by a fixation-based approach. If a user is looking at a picture of a face, is the attention focused on the entire face, the mouth, or just the bottom lip? From this example, it can be seen that it is difficult to determine user attention without a high-level analysis of the data. Eye tracking data contains other information useful for determining user attention, rather than just fixation locations. Task-oriented looking, for example, provides several eye motion cues that can only be understood when considering where the eye came from and the location to where it moved.

4. Solutions in Eye Patterns

Eye motion patterns are at the center of this research. Patterns of eye movement are comprised of a sequence of points representing the locations of the eye fixation points[2] over a certain interval of time. While research has been performed on patterns of object selection to infer user intention and state (see Section 2.5), this work explores a new direction: the use of *patterns of eye fixations*. There are several advantages that can be gained from analyzing patterns of eye motion in eye tracking interfaces. The technique of analyzing eye movement on the pattern level can have three significant effects on current eye tracking systems that this section will propose and discuss. Such a technique can offer speed, reliability and a better understanding of user attention. These three effects directly address the three problems outlined in Section 3 and are individually discussed below.

4.1 Speed though Pattern Identification

Pattern identification in eye motion data can increase selection speed. Identifying a pattern of eye motion can be much quicker than detecting a sequence of object selections. As pointed out above, the task of using eye fixations as a means of selection can take an unpredictably long period of time depending on how good the system accuracy and system calibration is (see Section 3.1). This can greatly delay a system's response time.

The use of a pattern of fixations as a means of selection bypasses the need to select individual objects with fixations and thus can dramatically speed up selection time. Figure 3.a. displays a fixation path using a normal fixation-based method that requires a

fixation to land on a target for a specified period of time. Figure 3.a. shows the need for several fixations per object selection when using fixation for selection. Figure 3.b. shows the use of eye fixation pattern for selection, which requires fewer fixation points to identify a selection. Because selection through pattern does not require many fixation points, selection speed for such a task is dramatically improved with little if no cost to accuracy (see Section 4.2).

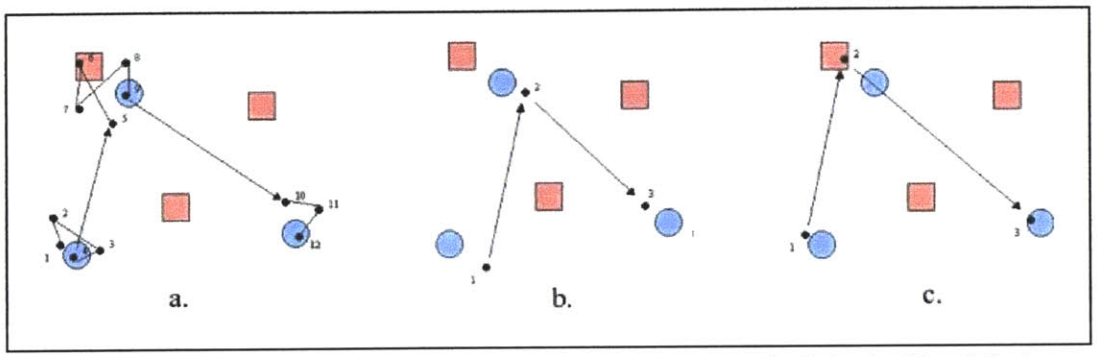

Figure 3. Three different eye fixation paths are shown for the same task of selecting blue circles in the image. *a.* selection of three objects by fixation *b.* selection of three objects by pattern of fixation *c.* selection reliability through pattern correlation

4.2 Reliability through Pattern Correlation

This research proposes that the technique of analyzing eye movement on the pattern level can improve reliability in eye tracking interfaces by increasing accuracy in target acquisition. In order for a pattern to be recognized, a very characteristic path must be taken by the user's eyes. Such a requirement can lead to a specificity that is hard to produce by either accident or luck. A system that is designed to listen for eye patterns waits for the user's eyes to move in a certain pattern before an event or action is initiated. This offers a much more reliable response than a system that looks only at the location of the user's gaze to initiate action.

There are several features of an eye movement sequence that can make pattern identification more reliable than fixation identification. The distance and vector angle between two points in an eye movement sequence are both attributes that can be used to help validate a pattern. Even the information concerning the location of where the eyes start and end can be used to help confirm a pattern. Identifying these features can provide data redundancy and correlation in noisy data from eye tracking systems. Figure 3.c. shows how pattern correlation can improve the reliability of a selection. This figure shows how the task of selecting blue circles can be distinguished from the task of selecting red squares based on the pattern of the eye fixation path. Using a pattern-based approach, a system can examine several validating factors to establish consistency in interpreting a user's intent, which ultimately improves the reliability of the interface.

4.3 Understanding User Attention through Pattern Interpretation

A system using an eye pattern based approach can better understand the concept of user attention. As discussed earlier, there are several problems with the way fixation-based eye tracking systems determine user attention (Section 3). Without looking at sequences of eye motion, it is difficult to appreciate attention on a complete level. Through examination of eye motion at the pattern level, the scope of a user's interest/attention can be better determined and identified.

The scope of user interest/attention is not something that is adequately addressed in current eye tracking systems. A traditional eye tracking system generally approaches the task of identifying user attention based on eye fixations. While the direction of the gaze usually points to the object of interest, this is not always the case. Several fixations within

a particular area might indicate the user's interest in a single object in that location, or it could also be indicative of interest in a couple smaller objects. A system that has knowledge of the objects in the image and uses a pattern-based approach can better determine if the user is interested in a face, or specifically the nose on that face. By looking at eye tracking data in aggregate patterns, the data can be processed at a higher semantic level. The points of eye positions while giving little meaning themselves, can be grouped into patterns that have relevance to what the user is really looking at and what the user is concerned with.

Eye patterns can also give a better indication that a user has in fact given attention to a particular object. A system that can combine the history of a user's gaze with information about the objects in an image can build a better model of attention. The current location of a user's eye-gaze alone has proven insufficient for determining attention but the analysis of how the user's eyes moved in a given time period gives a much more complete picture. With this technique, the problem of distinguishing meaningful attentive vision from idle looking will be easier to approach.

Patterns of eye fixation can directly reflect user task. This represents a new area not emphasized by current eye tracking work which has primarily focused on patterns of object selections. Salvucci (1999) proposes similar work using fixation tracing to facilitate eye movement analysis to infer user intent at the fixation level. This work helps to better infer intended fixation location from recorded eye movement (see Section 2.5). Work performed by Edwards (1998) distinguishes eye movement into three mutually exclusive categories of behavior: searching, knowledge movement, and prolonged searching. From these characteristic patterns of eye movement, inferences can be made

about user intent. This research investigates how aggregations of eye motion patterns can correlated to contextual user attention, such as user task. A computer user's eyes move in a specific way across the screen that is characteristic in part of the type of task, whether writing a paper, browsing the web, searching for a file, checking email or launching an application. Patterns promise the potential of helping eye tracking systems begin to understand the user on a much higher-level.

5. Research Overview

This research aims to demonstrate the value of interpreting eye motion through eye fixation patterns. Current eye tracking interfaces have difficulty in determining user intent from recorded eye movement. This is due to the approach of identifying selection through local points of recorded eye fixation. There are several problems with this fixation-centric approach concerning issues of selection speed, system reliability and the understanding of user attention. This research proposes a solution in a pattern-based approach in interpreting eye motion data. Experimental investigation is performed to evaluate and compare these two approaches. For the purposes of this research, an eye tracking interface system called InVision is built that incorporates a pattern-based analysis of eye motion. This research uses InVision to build two separate applications, EyeTester and Kitchen InVision, each demonstrating aspects of the stated research objective.

EyeTester quantitatively investigates the performance of an interface that is able to process data on a pattern level and is compared to fixation-based interfaces. Two related aspects of performance are measured for each interface: speed and accuracy. Because of the wide range of eye tracking algorithms that exist, the importance of the experiment performed lies not within the specific data obtained, but in what the results say about the role patterns can play in selection accuracy and selection reliability.

Kitchen InVision is an experimental interface that seeks to demonstrate the ability of a pattern-based interface to better understand the direction of a user's attention by

interpreting a user's eye motion on a high-level. Specifically, this interface is receptive to task in the context of a kitchen.

The following sections will describe the work performed in this research. First, the InVision system is described, next the EyeTester work is covered, and last, the Kitchen InVision research is described.

6. InVision

The InVision system is an experimental interface designed to listen to a user's eye patterns to help better understand and identify user attention. The system uses a pattern-based approach to analyze the eye tracking data. By creating this pattern-based interface, the hypotheses proposed by this research can be examined. InVision sets up a platform on which the remaining research is based. It is hoped that this interface will demonstrate the value of looking at eye fixations as aggregate patterns instead of independent and disjoint occurrences which is common in current eye tracking techniques. It is also hoped that this work will encourage a better understanding of eye patterns by providing an interface that can be used across different projects. The purpose of this research is not to create something that will replace current eye tracking interfaces. It is meant instead to be an investigation as to the value of processing eye motion on the pattern level. The specific use of InVision in the EyeTester and Kitchen InVision projects are described later (see Sections 7, 8). This section discusses the InVision system from a technical standpoint. First, the BlueGaze/BlueEyes eye tracking system that interfaces with the InVision system is described. Next, the InVision system's design criteria and implementation are discussed.

6.1 BlueGaze and BlueEyes

BlueEyes is an eye tracking camera system[3] that is being developed by IBM at Almaden Research Center. A BlueEyes eye tracking camera was assembled at IBM Almaden and brought to the MIT Media Lab for the purposes of this research. While a

brief description of the camera used is given below for the purpose of documentation, this paper does not discuss the details of the camera hardware and camera setup as they are not a focus of this research.

The BlueEyes eye tracking system uses a combination of pupil detection and corneal reflection to estimate the location of a user's gaze on a screen. The technique of reflecting infrared light off the eye is one used by several commercial vendors of unobtrusive eye tracking technology. The BlueEyes camera determines vision location by using a new approach to this technique. Two near infrared time multiplexed light sources are synchronized with the camera frame rate. The system operates at 30 frames per second.

One source is on-center, the other, off-center with respects to the camera's optical axis. This allows for a reliable means of pupil detection.

Software for the IBM BlueEyes eye tracking camera, called BlueGaze, was also obtained through IBM Almaden Research Center. BlueGaze is an application written in MFC C++ that handles the image processing from the BlueEyes camera. This application interfaces with a video capture card that in turn, interfaces with the BlueEyes eye tracking camera. BlueGaze is also responsible for the calibration of the BlueEyes camera. With slight modifications, the BlueGaze code is adapted to interface specifically with the InVision system.

6.2 Design Criteria

The set of design criteria for the system is provided to give the reader the design goals for the system. The criteria have provided a constant set of guidelines for the

system, influencing the InVision project at all stages from the initial design to the actual implementation. This section lists the criteria and then discusses each briefly.

Design criteria for InVision system:
1. demonstrates the value of a pattern-based interface
2. creates an unobtrusive interface
3. creates an intuitive and natural interface
4. interacts with the user in real-time
5. allows extensibility for other projects

The first design criterion for the system places an emphasis on exploring a new interface as opposed to necessarily aiming to create a better eye tracking interface. As it turns out, the ideas in the InVision system offer traditional eye tracking interfaces a more powerful and complete way to look at eye motion data.

The second criterion, that the interface is non-obtrusive, is not necessary for the technology or the demonstration of the ideas. It is believed that a non-obtrusive eye tracking system provides a more compelling demonstration of what the analysis of eye patterns can reveal about a user. A non-obtrusive system is also better suited for non-command (Nielson 1990) and context-sensitive interfaces that transparently sense a user's eye motions and responds.

Related to this is the third requirement, that the system be intuitive and natural. This is necessary for creating non-command interfaces that can help understand user cognitive state.

The fourth criterion states that the system must be capable of interacting with the user in real-time. Speed is one of the biggest advantages eye tracking technology has to offer to human computer interfaces. Ware and Mikaelian (1987) show that simple target selection and cursor positioning operations can be performed twice as fast using an eye

tracking interface than using a mouse. If an appreciable delay in system response is incurred due to the incorporation of eye pattern analysis techniques, this removes much of the appeal of eye tracking interfaces.

The last criterion is that the system be extensible to other projects. The system was designed to support a wide range of variance from project to project. The decision to write the InVision interface in Java relates to this last criterion as well. The extensibility of the system allows the EyeTester and Kitchen InVision to be easily constructed, both of which build on top of the InVision system.

6.3 The InVision Pattern-Based Approach

A description of the pattern-based approach used by the InVision system is provided. It is referred to as an *approach* rather than an algorithm or a technique since it describes a general method for defining a pattern of eye fixation within a set of eye-gaze data. The pattern-based approach uses a measure of both angle of approach and distance traversed between two consecutive fixations in order determine the probability that two fixations belong to a pattern. Over a sequence of several fixations, the probability of a pattern being incorrectly determined is low since several points provide collective reinforcement. This topic is discussed in Section 4.2, a section that discusses how reliability can be provided through pattern correlation. The angle of approach between two fixations is measured and compared with each angle in the set of angles between object pairs in the interface. If the value of the angle of approach is close (within a specified range) to the angle between one of the object pairs, the distance between the two consecutive fixations is measured and compared to the distance between that

particular object pair candidate. If this distance falls within a designated range, then the next connecting angle of approach is measured. The process repeats until either a pattern is recognized or the system determines that no pattern in the interface exists that fits the angle and distance characteristics for that particular set of fixations.

6.4 Implementation

This section outlines the technical detail of the InVision system. This section first describes how the components work together at the system level, and then provides further detail of the each of the individual components in the system. The value of this work lies in the concepts and techniques used in the research, rather than the specific implementation of InVision. Therefore, this paper is written in a level of technical detail aimed to appropriately convey these ideas. Since it is not this paper's intention to describe a specific implementation of a pattern-based eye tracking system, neither code nor pseudo-code is referenced here. This paper seeks to outline the InVision system in adequate detail such that the steps of this research can be retraced whether for the purposes of adding on to this work, designing the architecture for a new system, or even for understanding how eye patterns can be examined and utilized by such a system. First a detailed overview of the entire system is given to describe how all the components work together. This description gives enough technical detail for a user who wishes to understand the ideas behind the research. Next the major components of the system are individually described. Depending on the reader's level of interest regarding the technical aspects of the project, this section may be skipped.

6.4.1 The System

This section outlines the InVision system architecture. First, a description of the problem is given. Next the modules of the system are examined, in the context of how each fits into the system.

In order to understand the architecture of the InVision system, it is important first to have an understanding of the problem that the system is designed to solve. The InVision system takes coordinate data from the user representing points of localized attention in the interface. There are two types of possible coordinate data from the user. The first type of coordinate data represents the location of the user's eye gaze on the InVision interface. This eye motion data comes from the BlueGaze application discussed earlier. The second type of coordinate data represents the mouse cursor location on the interface; this allows the user to simulate eye motion through mouse cursor movement. This data represents the location of the mouse cursor on the interface. Data is also collected about the objects in the interface, such as size, location, and state. A model combining these two sets of data is assembled and the data is analyzed for patterns of eye movement. InVision divides the task of processing the data into two separate levels, the first being a low-level analysis of the raw data, the second being a higher-level interpretation of the analyzed data. This two level system design allows the needed flexibility to experiment with and change the behavior and response of the system on a per project basis. It is an architecture that from the start is conscious of separating the task of recognizing patterns from the task of attributing meaning and response to these patterns. Different applications for eye pattern recognition systems each might have to recognize patterns that are specific to the particular application. Patterns of eye motion

are identified and aggregated into larger patterns until the system can deliver an interpretation with relative confidence. These interpretations are reflected in the changing visual state/appearance of the interface.

The InVision system architecture is now described by component in an order that reflects how data passes through the system. When appropriate, the components are explained both in relation to other modules as well as to the system.

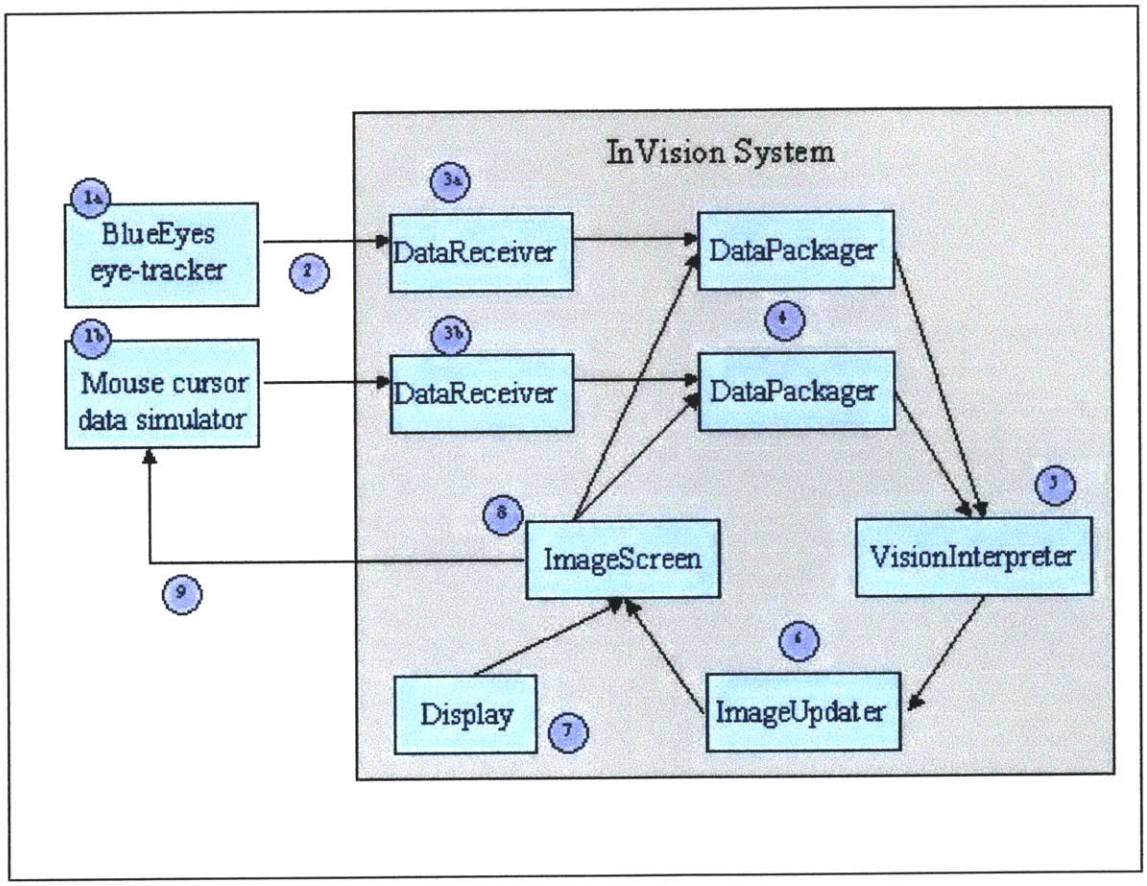

Figure 4. A data flow diagram of the InVision system (component part numbers referenced in text).

Figure 4 shows a data flow diagram that is directly referenced in the following component description:

1. Coordinate data from the user is collected that can take one of two forms:
   a. the coordinate location of the user's eye on the interface (from BlueGaze application) or

b. the coordinate location of the mouse cursor on the interface (from an element in the ImageScreen component that records mouse movements on the interface).

2. Each set of coordinate data is passed onto a dedicated TCP/IP socket. All communication from the BlueGaze application to the InVision interface is handled over this designated socket.
3. The *DataReceiver* is responsible for intelligently managing the data coming in to the system from the socket. The InVision system makes use of two concurrent *DataReceivers*:
   a. A *DataReceiver* is used for the BlueGaze coordinate data, and
   b. a separate *DataReceiver* is used for the mouse coordinate data.
4. The *DataPackager* receives each set of coordinate data from the *DataReceiver*. The *DataPackagers* package each coordinate set with information regarding the current state of the objects in the interface.
5. The *VisionInterpreter* accepts entries from the *DataPackager* and builds a model representing the sequential history of where the user has looked and where the objects in the interface were at a particular time. The *VisionInterpreter* records and analyzes the raw data.
6. The *ImageUpdater* uses the specific context of the interface to interpret the analyzed data from the *VisionInterpreter*. It holds the responsibility of updating the interface based on the interpreted data.
7. The *Display* component enables user interaction with the InVision interface. The user can toggle mouse-mode or eye-mode to allow different forms of coordinate data, reset the interface, or execute other custom actions defined for a specific interface.
8. The *ImageScreen* displays the visual representation of the interface to the user. This component interacts with the *ImageUpdater*, as well as the *Display* component.
9. Coordinate data representing the mouse cursor position is passed onto the TCP/IP socket.

6.4.2 InVision System Components

The InVision system components are now described in order to provide technical details regarding the architecture of the system. Understanding the level of technical detail that follows is not required to understand the ideas presented in this paper. The following sections are provided to document this research and provide the interested user with further information and detail regarding this system.

VisionInterpreter

The VisionInterpreter plays an important role in the InVision architecture, representing the model of the system. It is responsible for both maintaining a record of the system data and analyzing this data on a low-level. Both responsibilities are described in further detail below.

The VisionInterpreter is responsible for keeping a record combining the eye data as well as the object data of the interface. Records are maintained in real-time of what the user looked at and when the user looked at it. The history of the system is stored in a sequence of frames where each frame represents the state of the system at a particular time. A frame holds information regarding the state of the objects in the interface as well as the location of the user's eye on the interface. In this way, a sequence of frames together represent the various changing states of the system over a period of time. The VisionInterpreter maintains the history record at a constant size which allows manageability of the data. With the record of the user's eye position at a given time along with information about the state of the interface at this particular time, the VisionInterpreter has a very complete set of information regarding the sequence of events. By building a record of this data, algorithms can be run which identify, analyze and interpret patterns of eye motion.

This leads to the VisionInterpreter's second responsibility which is to identify and analyze patterns in eye motion from the recorded data. As the VisionInterpreter stores the data from the system, it also updates a set of records that are used for identifying patterns of eye motion. The VisionInterpreter uses these records as well as the system's history record to analyze the raw data. Analysis of the data includes the recognition and

identification of patterns and aggregating groups of patterns into larger patterns. In this way, the VisionInterpreter is able to respond to system queries regarding the analysis of different sets of data.

DataReceiver

The DataReceiver obtains the coordinate data from the user and makes it accessible to the InVision system. The coordinate data can take one of two forms: data from the eye tracking system representing the location where a user is looking, or the data from the mouse cursor position on the interface screen. This component manages the processes involved with receiving the data and allowing the system to access it.

The DataReceiver handles all the details of the low-level communication from the eye tracking software to the InVision system across a TCP/IP socket. First a communication socket is opened that waits for a connection to be made from an eye tracking application. This module expects sequences of integers in pairs on the communication socket representing the coordinate data. A predetermined communication protocol is used to allow data to be passed over the socket. The DataReceiver manages the communication socket intelligently, flushing stale data from the socket if there is a buildup of data in the socket. This is necessary both to ensure that only the most recent data gets used in the analysis and to remove lag in the system response.

The DataReceiver's second responsibility is to manage how data is accessed by the system. Special control is needed to manage how data is received from the socket and later used by the system since the process of obtaining data and the system's process for accessing the data are asynchronous. Synchronization is necessary in order to keep the

data paired correctly as coordinates.

DataPackager

This class is responsible for packaging the different data collected by InVision in a form that can be recorded by the InVision system. There are two sources of data being collected in this class: the position coordinate data, and the object data from the interface. The data that is collected regarding the interface objects includes such information as the object's location, size, state, whether it is visible, and whether it is active. Data from the two sources are packaged together in an entry such that each set of coordinates is paired together with information regarding the state of the interface at the time. Each entry represents both the state of the image, as well as the location of the user's immediate attention at a specific time. The DataPackager constructs each entry and add it into the InVision system as the required data becomes available.

ImageUpdater

The ImageUpdater is responsible for determining how to update the visual appearance of the interface based on user interaction. The interface is updated based on the user's past and current eye motion. The ImageUpdater interprets the analyzed system data on a high-level in order to decide how to best update the interface.

ImageScreen

The ImageScreen is the visual interface of InVision, and handles the graphics and visual representation of the interface. This component is responsible for visually

representing objects in the interface display in a way that reflects how the user is interacting with the interface. The ImageScreen provides a basic structure that can manage the graphics of the system. It properly loads the images used in the system to ensure that the animation of the system executes as intended. The ImageScreen module also handles the mouse interface system that can simulate eye movement across a screen based on mouse-cursor location. This defines what actions are taken when the user's mouse is pressed and moved.

Since each project using the InVision system has a separate function, each will require a different graphical interface. ImageScreen is a component that allows a developer to easily create visual interfaces that can be used with the InVision system.

ImageObject

The ImageObject component is used to create each of the objects in the interface. The ImageObject provides a standard way to define the objects for any interface that uses the InVision system. When all interface objects are defined through this structure, InVision is able to interact with the interface objects from project to project in the same manner. This ImageObject gives the developer a structure to follow when defining the objects of a particular interface and when planning how the objects should interact with one another. This structure holds information pertinent to the specific object such as images, states, information about what areas on the object are active in the interface, and certain attributes such as whether the object is visible or whether the object is active. When using ImageObject to define an object, custom code can be included as needed to further define the object's relationship to other objects or to the interface.

Display

The Display component is responsible for how the entire InVision system is displayed. It manages the InVision panel that is displayed, specifying what control panels for the system exist and how and where the interface is painted.

7. Evaluating Selection Performance, the Eye Selection Test

The Eye Selection Test is an eye tracking interface designed to test user selection performance across different interfaces for purposes of comparison. This experiment demonstrates how a pattern-based approach can improve speed and reliability in an eye tracking interface in the task of identifying user attention. An experiment is performed to evaluate the pattern-based InVision interface in comparison to a fixation-based interface. The objective of this experiment is to address the first part of this research's hypothesis: to quantitatively investigate whether a pattern-based analysis can improve the reliability and speed of an eye tracking interface. Object size, while not the only measurable independent variable, is one of the biggest factors influencing selection performance. For this reason, selection speed and accuracy is measured for each interface over the size of the objects being selected. This provides a level means of comparison across different interfaces. The following sections outline the experiment performed, present the experiment results, and finally discusses the analysis of the results.

7.1 Design

There are two features that are desired in this experiment. This set of criteria is used to meet the objectives of the Eye Selection Test and are listed below:

Eye Selection Test Design Criteria:
1. Relevance and applicability addressed on a broad level
2. Results reflect a fair evaluation of interface ability

An experiment is desired whose scope of relevance and applicability is broad. This enables general conclusions to be drawn, rather than conclusions that are only pertinent

under very specific conditions. The aim of this experiment is to draw general conclusions regarding the performance of a pattern-based interface in comparison to a fixed-based interface. Such a criterion allows the experiment to demonstrate the general value of using a pattern-based approach.

A related criterion is for the results of the experiment to reflect a fair evaluation of the interfaces ability. How does one fairly measure reliability and accuracy in an interface system? Benchmarking these interface system proves to be a difficult task since specific scenarios exist that allow one system to perform well over the other. There are situations where the use of one approach is more appropriate over the other, or where a combination of the two approaches can even be used. There are several factors that affect an interface's performance, not just the algorithms used in the system. Some of the variables hypothesized to affect performance are: user ability, number of objects, object position, and size of objects. By randomizing these variables, the experiment is able to examine an average spread of data for target acquisition accuracy.

7.2 Experiment

The Eye Selection Test displays a sequence of trials each consisting of circular targets on the subject's screen. Targets appear three at a time in random locations on the screen (see Figure 5). Target size is randomized across trials but all the objects in the same trial are of equal size. The subject is instructed to select each of the three targets as rapidly as possible when the targets appear at the beginning of the trial. Selection, defined in the context of this interface, is equivalent to target acquisition. When an individual target has been selected, it disappears, and after the three targets in the trial have all been

selected, a new trial is displayed. If a trial is not completed in a designated time, it is skipped. The trial begins when the user selects the first target and ends when the last target selection in a trial is completed. The experiment records whether the targets have been successfully selected along with the time duration of the trial. The number of off-center fixations, or fixations that don't select a target, are recorded and are used to reflect the relative inaccuracy of the selection process. After the subject has completed the series of trials, data analysis is available. The data from each trial with the same object sizes is combined for the purposes of analysis.

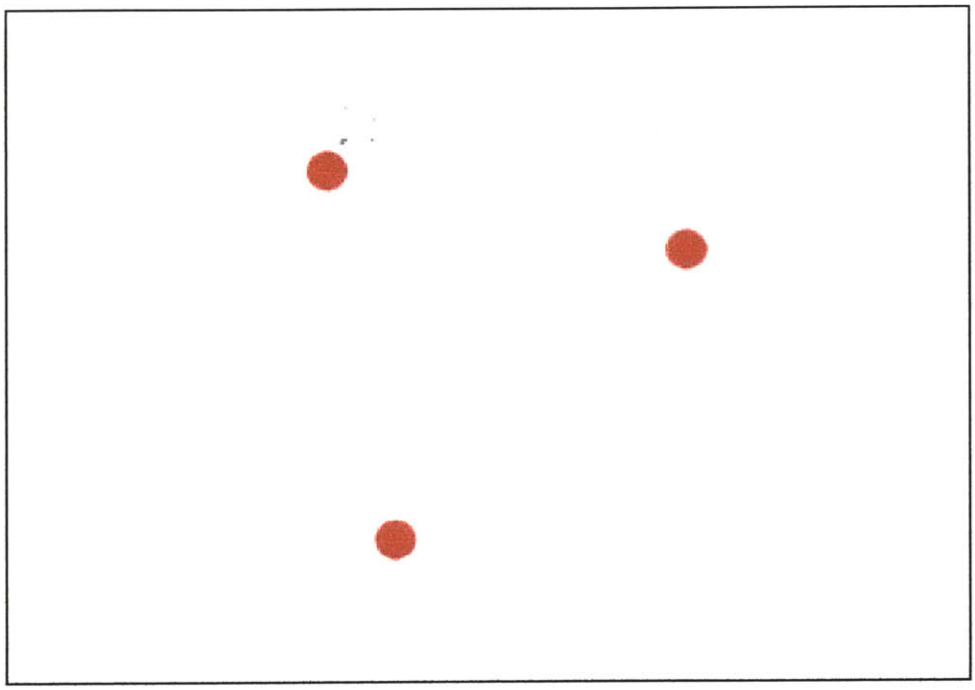

Figure 5. The Eye Test Selection Experiment

The InVision system's pattern-based approach (see Section 6.3) is compared to a system that uses simple fixation as a means of target acquisition, an approach that looks simply at whether a fixation falls within an object. While better fixation algorithms exist to recognize the intended location of off-center fixations (such as one that chooses the

closest target to the fixation), this fixation-based approach is chosen for use as a base level comparison. For the actual experiment, 5 tests each composed of 500 trials were run on each interface. Each selection test used a random target object size between 5 and 150 pixels and placed the target at a random location on the screen. Through this experiment, selection accuracy is compared across object size between a fixation-based interface and the pattern-based InVision system.

7.3 Results

The selection speed and accuracy of the InVision system is compared with the selection speed and accuracy of a fixation-based system. The results from the each trial are combined and summarized below in the following sections.

7.3.1 Speed Comparison

Data regarding trial times is collected from the experimental runs, combined and then summarized. In order to compare the two different sets of data, the time recorded for each trial is divided by three, the number of targets per trial. This gives a representation of the selection time per object for a certain object size using a particular interface. The selection times recorded per experimental run reflect an average of the selection times across all trials of a particular object size. The selection times for each experimental run is plotted across object size for both interfaces and the results are summarized in Figure 6. A best-fit line is drawn through the two data samples.

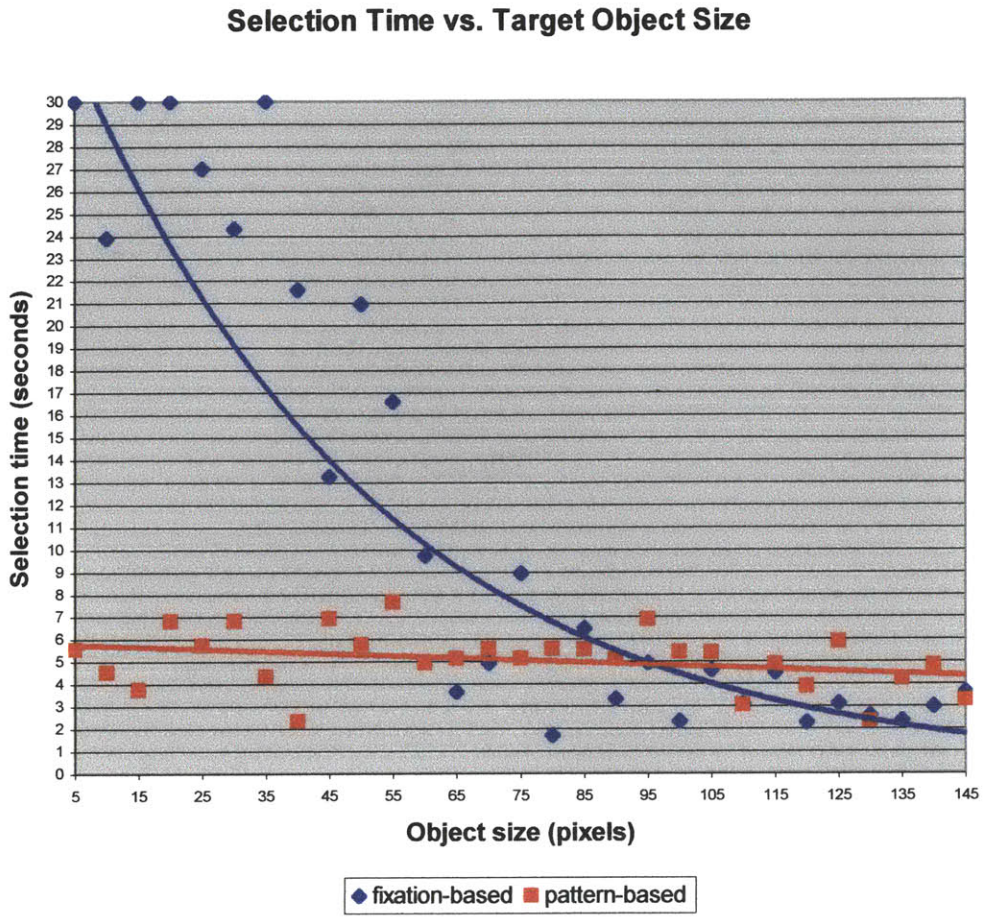

Figure 6. Selection Time vs. Target Object Size for Fixation (blue) and Pattern-Based (red) Approaches

7.3.2 Accuracy Comparison

Selection accuracy is used as a measure for system reliability. Data is taken from the same experimental runs used in the speed comparison analysis performed above. Selection accuracy for a particular object size is measured by dividing the number of target objects within the trial by the total number of fixations recorded in the trial. Similar to the speed comparison analysis performed above, the data collected from each experiment for each object size is averaged across trials. The percent selection accuracy

for each experimental run is plotted across object size for both interfaces and is displayed in Figure 7. A best-fit line is drawn through the two data samples.

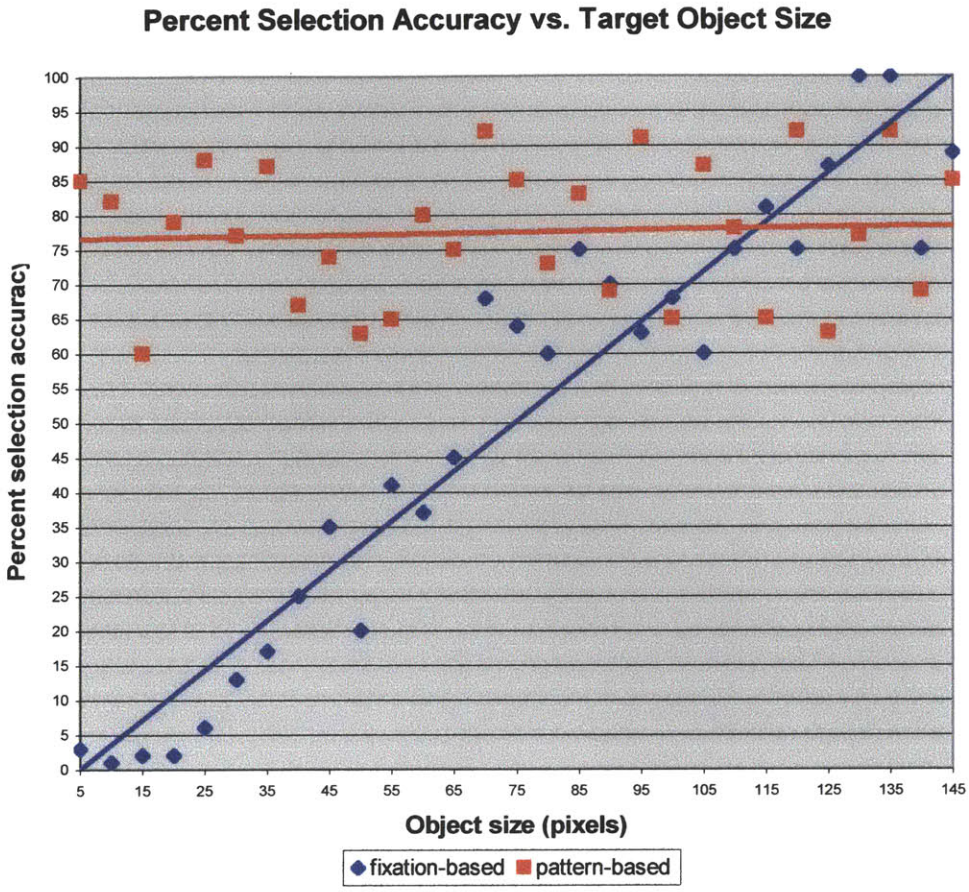

Figure 7. Percent Selection Accuracy vs. Target Object Size for Fixation (blue) and Pattern-Based (red) Approaches

7.4 Discussion

The discussion of the results is separated into two independent discussions to separately address the results from each of the two experiments outlined above. A discussion regarding the interface speed results is first given, followed by a discussion of selection accuracy.

7.4.1 Speed Comparison

A discussion on the comparative speeds first requires an understanding of how the selection time is broken down. From observation, there appears to be only one main source that adds to selection time in this interface: the time required to fixate on a specific location. The selection time, t_{select} is approximately equal to fixation time multiplied by the number of fixations necessary to select an object. This is expressed as:

$$t_{select} = t_{fixate} * f_{target}$$

where:

f_{target} = number of fixations needed to select a target
t_{fixate} = time required for eye to fixate on an object
t_{select} = the average time required to select an object

The number of fixations needed to select a target is determined as the total number of fixations recorded in a particular trial divided by the number of target objects selected in the trial:

$$f_{target} = f_{trial} / n_{targets}$$

where:

$n_{targets}$ = number of target objects in trial
f_{trial} = number of fixations recorded throughout entire trial

It is apparent that the data set representing the fixation-based approach requires a rather large number of fixations compared to the pattern-based approach. For the fixation-based technique, the number of fixations need to select an object is non-linear and reflects the calibration and inaccuracies in the eye tracking system. In a fixation-based approach, several fixations are required before a fixation falls on the target object. At small object sizes, the fixation-based approach requires a large selection time. This is due to the fact that as object sizes decrease, objects becomes harder to select. However as object sizes increase, the selection time approaches a constant. In this case, large objects

are easy to select and just require the time of one fixation per selection. For this reason, as object size increases, the number of fixations needed to select a target approaches 1:

$f_{target} \rightarrow 1$
$t_{select} \rightarrow t_{fixate}$

An examination of the data set from the pattern-based approach shows a relatively constant selection time regardless of object size. This is due to the fact that selection is initiated through the pattern of the object locations, rather than being a function of the selectability of the individual objects. The time required to select an object is approximately equal to t_{fixate}, which means that in order for a pattern to be selected, each object in the pattern requires approximately just one fixation. This approaches perfect eye tracking (where the fixation always lands on the object).

The significance of these results is that the pattern-based approach is able to remove the unpredictability and non-linearity of selection by not selecting through fixation but through pattern.

7.4.2 Accuracy Comparison

The results for the accuracy comparison show that the InVision interface performs at a much higher selection accuracy than the fixation-based interface that is used for the experiment. Figure 7 shows a graph of the selection accuracy of the interfaces while object size varies. This graph shows that InVision performs better across all object sizes and performs significantly better when the object size is small, maintaining a high level of selection accuracy where the fixation-based system becomes very inaccurate.

On average, the InVision system's selection accuracy is much higher than the fixation-based system's accuracy. At large object sizes, there is little difference in the measured selection accuracy. At small object sizes, there is a marked difference in selection accuracy performance. In fact, objects with a size smaller than 20 pixels were practically impossible for the fixation-based technique to select. The fixation-based results show very poor accuracy since it becomes more difficult to select an object the smaller it becomes. Pattern-based interfaces however still show high selection accuracy under these conditions. Both sets of data show that the task of selection becomes harder as the object size decreases, which is logical.

The conclusions drawn from these results point to an increase in selection ability and accuracy using the InVision pattern-based techniques compared to an ordinary fixation-based approach. Knowledge of where the eye came from, specifically what the angle of approach and the distance traversed is, can improve an interface's ability to select target objects.

The InVision system is also more adept at distinguishing between close objects than other systems. InVision correctly distinguishes the target object from two objects that are relatively close on the screen at a higher frequency than does the fixation-based technique that is examined. This observation is logical since the InVision system is using additional information to help correctly place a fixation, not just the location of the fixation itself. If the user's fixation is measured halfway between two objects, the InVision system uses the angle of approach to the fixation to help determine which is the target object. This sets InVision apart from other target acquisition algorithms such as algorithms that use a shortest distance heuristic between a fixation and available objects.

8. Kitchen InVision

The Kitchen InVision project addresses another aspect of this research's thesis: demonstrating the use of eye fixation patterns to determine user intention and attention inherent within the pattern. The Kitchen InVision project is an example of a system that demonstrates these ideas in an eye pattern sensitive environment through a scenario that is convincing, tangible, and memorable. Previous work has been performed on inferring user intent from eye selection patterns, which follows a very similar direction of work. Previous research, however, focuses on patterns of selection points, rather than patterns of raw fixations, which is the focus of this project. Patterns of fixation, as demonstrated in this research, promise speed and reliability in interface response. The goal of the Kitchen InVision project is to investigate how a user's eye fixation patterns can be analyzed and interpreted to help a system understand a part of the user's cognitive state. This can be achieved by limiting the scope of the scenario to a specific context, in this case a household kitchen. This provides a context that is familiar to the user, offering a mode of understood interaction and behavior. This research is motivated in part by the inability of current eye tracking interfaces to adequately identify user attention. It also serves as an example of how patterns of eye fixation are inherent in the everyday interaction with the world. This research encourages the study of patterns of eye fixations by demonstrating that these patterns naturally exist and by showing that they can be used by interfaces to augment interaction ability.

8.1 Interface

The Kitchen InVision is a non-command interface that can interact with a user by *listening* to patterns of eye fixation. An image of a kitchen is displayed along with several items commonly found in a kitchen. Interaction is achieved by watching a user's eye motion, interpreting patterns of eye fixation and delivering a visual response reflecting a change in the state of the kitchen or one of the items.

The interaction across this interface from user to system is not necessarily a direct one. The project described does not employ direct control and manipulation but rather builds a non-command interface in which the system *responds* to the interpreted eye pattern rather than being controlled. Because of this, there are no command instructions provided on how to use the system. Providing instructions specific to the interface control implies a direct control mode of interaction between user and system, which as stated earlier, is not the intent of the project.

There are a couple defined tasks that can be recognized by the system: cooking a turkey, washing dishes, and unpacking food from a grocery bag. Each task is made up of a sequence of independent actions. For example, the task of cooking a turkey involves opening a cabinet, putting a tray on the counter, taking the turkey out of the fridge, putting it on the tray, sticking the turkey in the oven, cooking the turkey and taking it out at the appropriate time. Active objects in the kitchen function in much the same way as one would expect them to in real life; the fridge, oven, and cabinets can open, food items can be cooked or be placed in the fridge, dishes can be washed, etc (see Figure 8).

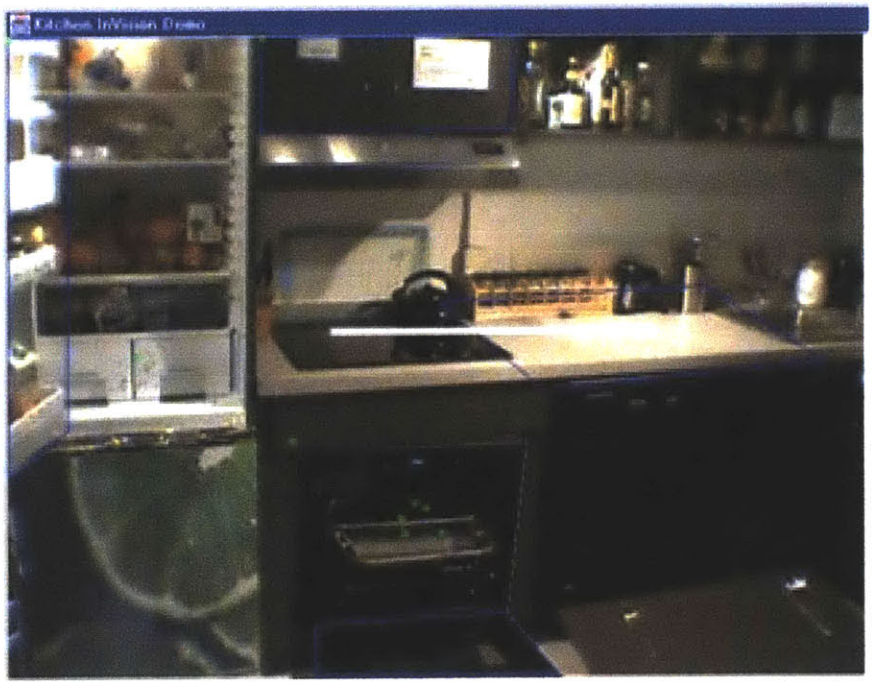

Figure 8. The Kitchen InVision Project

The Kitchen project depends on InVision to recognize several types of fixation patterns. Simple patterns such as detecting user attention on one or more objects are recognized by the system, as well as complex patterns involving higher-level analysis and interpretation of aggregate patterns.

8.2 Observations

Kitchen InVision is a project whose purpose is one of investigation and demonstration rather than experimentation. Unlike the Eye Selection Test whose results are quantitative, the results for the Kitchen InVision project are qualitative observations.

The purpose of the Kitchen InVision project, as stated earlier, is to demonstrate the value that the analysis and interpretation of eye patterns can offer in determining high-level user attention. Knowledge of where the user's eyes have moved in a particular

context provides a more complete understanding of user attention than merely identifying eye-fixations. The Kitchen InVision project is able to successfully demonstrate that an interface can use eye motion patterns to understand something about user cognitive state, in this case, task. The user's task can be identified based on the patterns of eye motion taken around the image. The user does not need to be given detailed instruction in order for the system to start identifying what task the user is trying to accomplish. During demonstration, the interface appears almost as if the system is reading the user's mind which begins to approach the high-level understanding of user attention that is desired.

8.3 Discussion

Several issues relating to the Kitchen InVision project will now be discussed. This section discusses the expectation and explanation of the observations, the limitations in the technique used, and finally, this research's value in the area of non-command and context-sensitive interfaces.

The Kitchen InVision project demonstrates how eye patterns can be used to interpret high-level user attention. The success of this demonstration is not entirely unexpected; it is logical that patterns of eye movement preserve data relating to a user's intention and attention. Where the eye comes from, what the eye has seen and when the eyes have moved all are factors that help understand user attention on more than just a localized scope. It should be stated that the concept of attention is a complicated idea that cannot be adequately identified by the eyes alone. However, as defined earlier, the scope of this research focuses on the eye's contribution to the state of user attention and

attempts to better understand what user attention is, and how to help identify it using data collected from eye motion.

There are several limitations and problems observed with this technique that should be noted. First is the problem of how to best interpret meaning from eye patterns, which is a challenge that the Kitchen InVision project investigates. For example, how can a system correctly identify when a user is casually looking at an image and when the user is engaged in a task? Analyzing eye motion on the pattern-level helps determine user attention, but still the best a system can do is to make an interpretation based on the information it receives. A second limitation observed in the Kitchen InVision system as a non-command interface, is that an experienced user of the interface, could learn how specific eye motions affect the system and modify his/her eye behavior in order to directly control the interface. The last observed limitation concerns the complexity of the system and the sophistication of the response. How is the boundary defined between what can and cannot be performed by the system? For example, in the Kitchen InVision system, the turkey cannot be stored in the cabinet.

The Kitchen InVision research, while a preliminary and very simple example of a non-command interface, is still relevant to the area of human computer interaction. It serves as an example of an interface that uses a combination of eye fixation pattern and contextual information as the means of identifying user task and intention. Previous endeavors to understand patterns of eye motion have emphasized the use of patterns of selected objects to make inferences about user intention, rather than patterns of fixations. A pattern-based approach can offer speed and reliability in the research of using patterns

to explain cognitive intent, especially in the area of non-command interfaces and context-sensitive environments.

9. Conclusion/Summary

This research has proposed the use of interpreting eye motion data through patterns of aggregate eye movement. A system called InVision is built which adopts a pattern-based approach to eye motion interpretation. InVision provides a platform on which interfaces using eye pattern analysis can be built. The abilities of a pattern-based approach are tested and evaluated by using the interface structure provided by InVision. Next, comparison benchmarking is performed between a pattern-based and a fixation-based approach. Finally an interface is created to demonstrate how patterns of eye fixation can be used to infer context-specific user intent. Results point to several advantages gained through the use of patterns, confirming the benefits of a pattern-based approach proposed earlier by this paper (Section 4). The three benefits gained from the use of eye pattern analysis, are:

1. Speed through pattern identification
2. Reliability through pattern correlation
3. Understanding through pattern interpretation

These will now be examined in the context of the results provided by the InVision project and the associated research. Conclusions regarding these benefits are now discussed. The paper concludes with thoughts on future directions for this work.

9.1 Speed through Pattern Identification

The speed of a selection or target acquisition through the use of patterns is very quick compared to that of regular fixation-based methods. The target selection times of a pattern-based and a fixation-based approach are measured and the performances of these

two interfaces are compared. The quantitative investigation shows a significant difference in selection time. For any size target object, the pattern-based approach to selection yielded a much quicker selection time in comparison to the fixation-based approach. For example, for objects that are 20 pixels wide, or roughly the size of a small button or a desktop icon, the pattern-based approach required an average of 500 ms for selection while the fixation-based approach required an average of almost 6000 ms seconds. While eye movement is fast, eye selection might not necessarily be fast. Using patterns of fixation for selection solves this problem not by proposing a new way for object selection, but by bypassing the object selection altogether. Selection through pattern identification is a powerful approach since it circumvents the problem of waiting for a system to register an off-center fixation as an intended selection. This improves response speed, since the number of fixations (and hence the sum of the fixation times) required per object in the pattern that is selected is reduced. This points to the conclusion that such a pattern-based approach allows selection time to come close to the speed of fixation time for every object selected, creating a dramatic improvement in interface speed.

9.2 Reliability through Pattern Correlation

Correlating patterns to eye-gaze data provides a powerful way to validate intended fixation location and ultimately user intent. The InVision system demonstrates the use of pattern correlation to add reliability in system response.

The intuition behind this demonstrated result is straightforward: if the objects in an interface are known, then there exists a finite number of locations that the eye will logically fixate on. More importantly to this research, if there are a finite number of

locations for logical eye fixations, then there is a finite number of ways of looking between these objects as well. The possible patterns of eye motion can be even further reduced to ones that make sense given the context of the situation.

An experiment is performed that compares the selection accuracy of the pattern-based approach in the InVision system to a fixation-based approach. Overall, the results show a significantly higher selection accuracy with InVision than with the fixation-based approach. Results show that the difference in selection accuracy between the two interfaces is dependent on object size. At large object sizes, selection accuracy for both interfaces is very good, both approaching 100% selection accuracy. Interface control however, becomes unwieldy in these conditions since at these large object sizes, selectable objects are several inches wide. At small object sizes, there is a marked difference in selection accuracy performance.

Given these results, one can conclude that there is an opportunity for patterns of eye motion to play an important role in system reliability. Context knowledge of whether a particular eye movement makes sense for a particular task can be used to infer user intent from eye patterns, a topic further discussed in the next section. Interpretation through patterns gives the system knowledge of the previous location of fixation, the distance traversed to the current fixation, as well as the angle of approach to the next fixation between the two fixations. By combining these validation features with context knowledge, InVision establishes consistency and reliability in interface response. InVision demonstrates higher accuracy in inferring the intended location of a recorded fixation than a fixation-based interface.

9.3 Understanding User Attention through Pattern Interpretation

The interpretation of eye fixation patterns can be used to understand user intention and attention. Patterns of eye movement preserve the structure of intention and attention of a user, a capability not present in a fixation-based approach. Because of this, analysis of eye motion on the pattern level can provide an understanding of user cognitive state. Through pattern interpretation, the InVision system is able to understand concepts such as the scope of a user's attention, as well as intended local attention. Concepts inherent in eye patterns such as task, searching, association and intention can be inferred from patterns of fixation as well. The scope of a user's attention can be better defined by a system that understands the history of a user's eye motion as well as what it means in the context of the scenario presented. The InVision system is more capable of recognizing and identifying user interest/attention by contextualizing fixations within both the scenario and the history of eye movement. Pattern interpretation also enables the InVision system to better distinguish between fixations in casual glance from those reflecting genuine user interest. The Kitchen InVision project demonstrates the use of the InVision project to identify, interpret, and respond to high-level user attention. Specifically, this research has investigated how eye patterns within a known context, can reveal information regarding user intention and task.

9.4 Future Work/Recommendations

The InVision research represents only a first step towards creating better eye tracking interfaces through analysis of fixation pattern. As such, there are several promising directions for future work. The goal of this research is to collect data and

preliminary results from which to draw initial conclusions and evaluate this pattern-based approach. It is hoped that the results and associated research provides motivation for the further study of this approach. Further exploration into this topic is needed, not to demonstrate its value (which this research has done), but instead to begin thinking about where patterns of eye fixation can be useful and how to begin incorporating such techniques into eye tracking interfaces.

A very general direction that could benefit from the implications of this research, is any area that is already studying the use of eye motion patterns through patterns of selection. There are three broad (and most likely overlapping) areas of possible attention whose benefits gained from pattern-based approaches can easily be identified. The first is the use of such techniques to quickly identify and then understand associations and relationships in localized user attention. This is an area which studies how to infer meaning from patterns of association for purposes ranging from customizing content to market focusing. The second is the use of patterns to quickly identify user task and intention. One can image intelligent tutorial scenarios such as flight simulator training application that had knowledge of why a user made a particular decision based on how the eyes moved about the instrument gauges. The third area is the use of patterns of eye motion in a context for social interaction. This is specifically applicable to the area of interacting with socially intelligent agents.[4] A pattern-based approach in this research area can help computers understand a person's high-level attention, offering a new means of interaction between human and computer.

To conclude, analysis of fixations on the pattern-level has been identified as a new approach for selection that can offer both better ability as well as new capability to eye

tracking interfaces. It is hoped that the research outlined in this paper will give encouragement to future research and development of eye fixation patterns.

End Notes

[1] Unless specifically cited, most of the content in the background section that is specific to the physiology of the human eye and eye motion comes from (Babsky, Khodorov, Kositsky, and Zubkov, 1975) or (Bruce & Green, 1990). Ideas and information that I believe might be of particular interest to the reader are individually cited (even though they may come from one of these two sources).

[2] In this research we look at patterns of fixation points, although eye movement patterns are not necessarily limited to fixation points alone.

[3] Readers who are interested in learning more about the IBM BlueEyes camera can reference the work at the following URL: http://www.almaden.ibm.com/cs/blueeyes/. Information can also be provided by email by writing to the following address: blueeyes-db@notes.research.ibm.com.

[4] This research work has been presented at the Fall 2000 AAAI conference on Socially Intelligent Agents in Falmouth, MA.

Acknowledgements

I would first like to thank my thesis advisor Ted Selker whose eccentricity has added balance to my life. He has given me encouragement throughout, and has carried the enthusiasm for the research when mine ran low. I thank him for his continuing confidence in me as well as my work.

I would also like to thank IBM Almaden for allowing me to use the BlueEyes camera and the BlueGaze code for my research. Without their generosity, this research would not be possible. Specifically I would like to thank Myron Flicker and David Koons for assisting me in building and setting up the BlueEyes eye tracking system. Myron has also been patient enough to provide me with support, helping me with the problems and questions I have encountered with the BlueEyes system.

Thanks go to Shumin Zhai at IBM Almaden for advice on how to evaluate the performance of the InVision system built through this research.

Finally, my thanks go to my friends and family who have kept me sane throughout my career at MIT.

References

Babsky, E., Khodorov, B., Kositsky, G., & Zubkov, A. (1975). Human physiology. Moscow: Mir Publishers, 182-214.

Barber, P. J. & Legge, D. (1976). Perception and information. London: Methuen, chapter 4: Information Acquisition, 54-66.

Blue eyes: Suitor [WWW Document]. URL http://www.almaden.ibm.com/cs/blueeyes/suitor.html (visited 2001, February 2).

Bruce, V., & Green, P. R. (1990). Visual perception: Physiology, psychology and ecology, 2nd edition. Hove, UK: Lawrence Erlbaum Associates Ltd.

Edwards, G. (1998). A tool for creating eye-aware applications that adapt to changes in User Behaviors. International ACM Conference on Assitive Technologies 1998. 67-74.

Engell-Nielsen, T., & Glenstrup, A. J. (1995, June 1). Eye controlled media: present and future state [WWW Document]. URL http://www.diku.dk/~panic/eyegaze/article.html (visited 2001, February 2).

Goldberg, J. H. & Schryver, J. C. (1995). Eye-gaze determination of user intent at the computer interface. In J. M. Findlay, R. Walker, & R. W. Kentridge (Eds.), Eye movement research: Mechanisms, processes, and applications (491-502). New York: Elsevier Science Publishing.

Jacob, R. J. K. (1995). Eye tracking in advanced interface design, *in* W. Barfield & T. Furness, eds, `Advanced Interface Design and Virtual Environments'. Oxford: Oxford University Press, 258-288.

Kahneman, D. (1973). Attention and effort. New Jersey: Prentice-Hall, Inc., Englewood Cliffs.

Nielsen, J. (1993, April). Noncommand user interfaces [Communications of the ACM, volume 36, no. 4]. 83-99.

Salvucci, D. D. (1999). Inferring intent in eye-based interfaces: Tracing eye movements with process models. Proc. CHI '99, 254-261.

Starker, I. & Bolt, R. A. (1990). A gaze-responsive self-disclosing display. Proc. ACM CHI '90 Human Factors in Computing Systems Conference, Addison-Wesley/ACM Press, 1990. 3-9.

Stern, J. A. (1993). The eyes: Reflector of attentional processes, CSERIAC Gateway IV(4), 7-12. A Synopsis by June J. Skelly.

Theeuwes, J. (1993). Visual selective attention: A theoretical analysis. Acta Psychologica 83, 93-154.

Ware, C. & Mikaelian, H. T. (1987). An evaluation of an eye tracker as a device for computer input. Proc. ACM CHI, GI '87 Human Factors in Computing Systems Conference. 183-188.

Yarbus, A. L. (1967). Eye movements during perception of complex objects, in L. A. Riggs, ed., `Eye Movements and Vision'. New York: Plenum Press, chapter 7, 171-196.

Zhai, S., Morimoto, C. & Ihde, S. (1999). Manual and gaze input cascaded (MAGIC) pointing. Proc. ACM CHI '99 Human Factors in Computing Systems Conference, Addison-Wesley/ACM Press, 1999. 15-20.