The Conductor Interaction Method: Interacting using Hand Gestures and Gaze

Dorothy Katharine Rachovides

Submitted for the degree of Doctor of Philosophy

Department of Computing
Lancaster University
UK

June 2004

Acknowledgements

I would like to thank all those who have helped me during the production of this thesis. In particular, I would like to thank my supervisor Alan Parkes, for his patience and his always-constructive criticism, which were crucial for getting this thesis completed. I would also like to thank Pete Sawyer for providing valuable feedback on my thesis and John Mariani for his feedback on the prototype. Special thanks to James Walkerdine and Peter Phillips for their valuable help and support over the past few years.

I have made a lot of friends over the years of my studies at Lancaster, in particular Zoë, Awais, Rouza, Adam, Paul, Rosemarie, Nelly and James H, thank you for making them enjoyable.

Finally I would like to thank my mother and the rest of my family who stood by my when I needed them.

Declaration

I declare that the work presented in this thesis is, to the best of my knowledge and belief, original and my own work. The material has not been submitted, either in whole or part, for a degree at his or any other university.

Dorothy Katharine Rachovides

Abstract

The Conductor Interaction Method: Interacting using Hand Gestures and Gaze

Dorothy Katharine Rachovides

Submitted for the degree of Doctor of Philosophy

Department of Computing, Lancaster University, UK

June, 2004

Over the past thirty years computers have increasingly become part of our everyday lives. Most humans have become computer users in one way or another, since many activities involve either the direct use of a computer or are supported by one. This has prompted research into developing methods and mechanisms to assist humans in interacting with computers (known as Human Computer Interaction or HCI). This research is responsible for the development of a number of techniques that have been used over the years, some of which are quite old but continue to be used, and some are more recent and still evolving. Many of these interaction techniques, however, are not natural in their use and typically require the user to learn a new means of interaction. Inconsistencies within these techniques and restrictions they impose on user creativity can also make such interaction techniques difficult to use, especially for novice users.

This thesis proposes an alternative interaction method, the Conductor Interaction Method, which aims to provide a more natural and easier to learn interaction technique. This novel interaction method extends existing Human Computer Interaction methods by drawing upon techniques found in human-human interaction. It is argued that the use of a two-phased multi-modal interaction mechanism, using gaze for selection and gesture for manipulation, incorporated within a metaphor based environment, can provide a viable alternative for interacting with a computer (especially for novice users). The model for the Conductor Interaction Method is presented along with an architecture and implementation for a system that realises it. The effectiveness of the Conductor Interaction Method is demonstrated via a number of studies, in which users made use of the developed system. The studies involved users of mixed computer experience.

Published papers related to this work

"***Technology, Tools and Creativity***", Dorothy Rachovides and Alan P. Parkes, Panel paper at the EPSRC panel session on "Creativity and IT" at the 4th Computers and Creativity Conference, Loughborough, 16 October 2002.

"***Metaphors and Multimodal Interaction***", Dorothy Rachovides, Zoë Swiderski and Alan Parkes, presented at COSIGN2001, 10-12 September 2001, published in the conference proceedings, pp. 125-128.

"***"Multi-modelling" of Multimodality Interaction***", Dorothy Rachovides, Zoë Swiderski, Alan P. Parkes Workshop on Computational Semiotics for New Media, University of Surrey, 29-30 June 2000.

"***The User as a Conductor in Multimedia Interfaces***", Dorothy Rachovides and Alan P. Parkes, CHI'98 Workshop on "Innovative Interface Metaphors for Visual Media".

TABLE OF CONTENTS

List of Figures

List of Tables

Chapter 1 – Introduction

1.1 Interacting with computers

Humans are increasingly using computers as part of their everyday lives. Computers can be used to facilitate human activities, such as work, communication and entertainment. Most of our activities today involve either the direct use of a computer by us or are supported by a computer.

Over the last 30 years much research has been carried out into developing methods and mechanisms to assist humans in interacting with computers (known as Human-Computer Interaction or HCI).

As computer technology has evolved so have the techniques used to interact with it, rendering some to be more successful than others. A key example of this has been the general shift away from command line interfaces to more visual interfaces such as the WIMP as a result of the increased resources being made available to interface designers.

The types of users have also changed. Whereas in the past computer users were typically programmers or scientists, in today's world the average computer user is a non-computer expert who may use a computer to support their work, but possess little knowledge of how the computer actually operates.

As computers are integrated more into our lives and used to perform a larger variety of tasks, humans have to learn how to use these new systems efficiently. There are many cases where the effort required to learn the use of a system outweighs the benefits of its use, (for example the interface of the 3D Max software is so complex, consisting of tabbed toolbars and multiple view ports, that in order for a user to be able to create a simple object such as a cube, a significant amount of learning is involved). The learning overhead that users are presented with can be considerable, which is one reason users are often reluctant to try new applications or interaction techniques.

A number of techniques have been developed for interacting with computers (Dix, 1998), some of which have been used for many years and continue to be used, while others are more recent and are still evolving. Early techniques included *command line interfaces*, of which a typical example is MS-DOS (Microsoft Corp), *menus,* such as those that use in MS-Editor (Microsoft Corp.) and *Form-fills* and *spreadsheets*. More recent techniques have included the use of (pseudo-)*Natural Language,* as seen in some web search engines such as Ask Jeeves (Ask Jeeves), *Question/answer* and *query dialog*, as used in on-line booking systems such as that of British Airways (British Airways). Finally there are the widely used *WIMP interfaces,* such as Mac OS (Apple Corp), *Point-and-click interfaces,* as demonstrated in Web browsers (Internet Explorer), and *Three-dimensional interfaces,* which are widely used in computer games and Virtual Reality environments (VR-VIBE).

1.2 Failings with existing approaches

Existing human-computer interaction techniques, such as those mentioned above, certainly can provide benefits for interaction as demonstrated by their continued use. Nevertheless these techniques do have certain failings, more specifically:

- *Existing interaction techniques are not natural and they typically require the user to learn a new interaction method.* As most techniques previously mentioned are significantly different from each other and from human-human interaction, users need to devote a significant amount of time to learn to use any given interface. Most of them heavily rely on memorisation of commands, which increases the learning curve.

- *Existing interaction techniques can be inconsistent.* This is an issue that arises with interfaces that are of the same type but have small differences. They can be confusing for any user, no matter how experienced.

- *Existing interaction techniques can restrict creativity and be over simplistic.* Existing interaction techniques are task driven and tend to be very structured towards that end. This approach can be very useful in guiding the user through a specific task, but can also be seen as an obstacle in the expression of the user's creativity.

This thesis provides the rationale for and describes an alternative interaction method that draws upon existing human-human interaction techniques. In particular it argues that a combination of gaze and gesture input modalities used in conjunction with a suitable interaction method, can provide a usable alternative for interacting with a computer. This interaction method aims to address the shortcomings exhibited by existing interaction techniques, and in particular aims to provide a more natural and easier to learn interaction method.

1.3 The Conductor Interaction Method

This thesis proposes a novel human-computer interaction method, the Conductor Interaction Method. This method allows users to make use of human-human interaction techniques, namely gaze and gestures, with which they can perform activities within a novel interaction environment. This environment is built around two metaphors that have been developed, that take into account the use of gaze and gesture interaction mechanisms. The method also makes use of a two-phased interaction process, where gaze is used for selection and gestures are used for manipulation.

A particular aim for the Conductor Interaction Method is for it to provide a more natural interface than existing interaction methods, and for it to have a low learning overhead. It is envisioned that this method will be particularly beneficial for novice computer users, who will be able to utilise interaction techniques they are more familiar with (for example, gaze and gestures), rather than having to learn totally new interaction methods. Areas where this method could be applied include creative domains, such as presentation creation, music, story creation, computer based learning etc., as well as control domains where the user may manipulate properties of an environment such as lights and sound within a house or night club, or cameras within a surveillance system.

The proposed Conductor Interaction Method is also supported by recent advances and reduced costs in the technology that can be used for capturing human-human interactions. The development of more affordable and sophisticated devices, whose cost was prohibitive in the past, has made it feasible to consider the adoption of Human-Human interaction methods into HCI. As a result, it is likely that there will be a growing move towards integrating aspects of human-human interaction into HCI, and consequently a need for research into new techniques and interfaces to support this.

The Conductor Interaction Method that is presented in this thesis is realised through the development of a system architecture and implementation. The architecture provides a structure for the development of a Presentation Conductor system, which enables users to create multi-media presentations using the Conductor Interaction Method. The implementation of this architecture allows for the interaction method to be demonstrated, as well as providing a mechanism for its evaluation.

1.4 Objectives of the work

The main objective of the research presented in this thesis was to investigate a novel approach to Human-Computer Interaction, which users would find useful, easy to learn and close to their everyday human-human interactions. This objective resulted in the development of the Conductor Interaction Method. This objective can be broken down into the following sub-goals:

- To develop a method for interacting with computers that utilises modalities that are typically used in Human-Human communication. The method would support the use of these novel interaction techniques with a specially tailored interface based on metaphors that users can relate to. Key requirements for the method would be for it to possess a low learning overhead and be easy to use.

- To demonstrate the feasibility of the method by designing and implementing a prototypical system, the Presentation Conductor, that would utilise it as the main method of interaction.

- To evaluate the Conductor Interaction Method through the Presentation Conductor. The evaluation should compare the method against existing interaction approaches to see whether it can be a beneficial alternative. The evaluation studies should involve a range of users with differing computer experience in order to provide a broad study.

1.5 Novel contributions of this work

The work presented in this thesis represents a novel approach to Human-Computer Interaction. The specific contributions of the work can be summarised as follows:

- *The Conductor Interaction Method.* This thesis presents a novel interaction method, which aims to provide users with a more natural way of interacting with a computer, and seeks to overcome some of the failings exhibited by existing human-computer interaction techniques.

  The presented method, the Conductor Interaction Method, makes use of human-human multimodal interaction mechanisms, namely gaze and gesture. The method also makes use of a novel interaction environment that is based on the Orchestra and Conductor metaphors.

- *Orchestra and Conductor metaphors.* This thesis presents two novel metaphors that have been developed in order to provide the user with an environment to interact in and a technique to support the interaction.

  The Orchestra metaphor provides the user with a 'stage' based environment in which he or she can interact by using a combination of gestures and gaze. The Orchestra metaphor graphically presents to the user the resources that are available for them to manipulate, as well as providing an area in which these manipulations are carried out.

The familiarity of the stage set-up gives the user the sense of expectation and helps in orienting them to the interface.

The Conductor metaphor is an interaction metaphor that is used in conjunction with the stage-based environment with the view of providing the user with an interface that is easy to use and understand, especially for inexperienced computer users. As the name suggests the user interacts with the resources depicted within the Orchestra metaphor in the same way that a conductor would interact with the musicians of an orchestra. The Conductor metaphor has been specifically designed for gaze and gesture based interaction methods.

- *An architecture and implementation for a prototypical system to demonstrate the Conductor Interaction Method.* This thesis presents an architecture for a prototypical system that demonstrates the Conductor Interaction Method. An implementation has also been developed that realises this architecture. The developed system, the Presentation Conductor, allows users to construct and display multi-media presentations by making use of the Conductor method. This system also forms the basis of the evaluation that has been carried out that aims to assess the usefulness of the Conductor Interaction Method.

- *Experience of using the method.* This thesis presents an evaluation of the Conductor Interaction Method via use of the Presentation Conductor system. The evaluation used a formative and qualitative approach, and involved users with varying computer experience performing a set of tasks with the system. The evaluation also assessed the impact different technologies could have on the effectiveness of the method.

1.6 Structure of the thesis

The rest of the thesis consists of six chapters.

Chapter 2 focuses on techniques and methods for interacting with computers. It begins by providing an overview of existing interacting techniques. Examples of each are provided and their shortcomings are discussed with respect to providing a natural interface for interaction. It is argued that to a large extent the problems that these approaches possess can be overcome by drawing upon the experiences of human-human interaction methods. In particular the chapter focuses on the use of gaze and 3D hand gestures within HCI. An overview of the more commonly used techniques for exploiting these two modalities is provided, with key technologies examined. Finally, the chapter discusses the benefits and feasibility of developing an interaction method that utilises such input modalities.

Chapter 3 examines nine key systems that utilise gaze, gesture or a combination of both as a means of interacting with the computer. An analysis of these systems is provided which indicates a number of issues that needed to be taken into account when designing the Conductor Interaction Method and developing the Presentation Conductor architecture. In particular the issues surrounding the Midas touch problem and how to support two-phased interaction were important factors that contributed towards the design of the Conductor Interaction Method.

Chapter 4 presents the Conductor Interaction Method, the interaction approach that is proposed by this thesis. Metaphors play a prominent role within the interaction method and after a brief overview of the role of metaphors in HCI, the Conductor and Orchestra metaphors are described. The Conductor Interaction Method is then described in detail and it is discussed how two-phased interaction, a more natural interface and reducing the user's learning overhead, have been taken into account. The Presentation Conductor system is introduced and an architecture for it is presented. Scenarios are used to illustrate the proposed operation of the system.

Chapter 5 discusses the implementation of the Presentation Conductor and how the Conductor Interaction Method has been implemented within it. The implementation realises the architecture that is presented in chapter 4, and allows users to create multi-media presentations via use of the Conductor Interaction Method. The Conductor and Orchestra metaphors are central to the implementation, with the Conductor metaphor supporting the interpretation of a significant number of hand gestures, and the Orchestra metaphor providing the appropriate environment in which to create multi-media presentations. The implementation has attempted to reduce the learning overhead by using gestures that are familiar to the user and can be readily associated with specific tasks.

Chapter 6 focuses on the evaluation of the Conductor Interaction Method via the Presentation Conductor system. The evaluation aims to assess the usefulness of the method as an interaction approach, and whether the choice of underlying technologies used can have an effect on its success. The chapter describes a formative and qualitative evaluation that involved a number of users, with differing computer experience, carrying out a set of tasks using the Presentation Conductor. The evaluation was spread over two stages. The first stage focusing on obtaining feedback on the Conductor method in comparison with existing interaction techniques. The second stage of the evaluation focused on obtaining feedback on the use of different technologies with the Conductor method. The results of both stages are presented and discussed.

To conclude this thesis, chapter 7 returns to the objectives that were set out in this chapter and discusses how they have been met by the work presented in this thesis. Finally the chapter discusses possible future developments to this work.

Chapter 2 – Interacting with Computers

2.1 Introduction

In order to develop a novel, more natural interaction method as is proposed by this thesis, it is first necessary to examine the existing state of work within the field of Human-Computer Interaction. This allows us to gain an understanding of existing HCI techniques, as well as to examine how these can be extended to include more natural techniques, such as gestures and gaze.

This chapter begins by identifying eight common HCI techniques. A general analysis of these techniques is provided that highlights their failings with respect to providing a natural means of interaction.

It is proposed that one way in which these failings could be tackled (at least to an extent) is by drawing on interaction techniques used for Human-Human interaction. The key aspects of Human-Human interaction are then discussed, with a particular focus on gaze and gestures. These two are discussed in detail examining their applicability in HCI, the main techniques used to capture them, and their general advantages and disadvantages.

The chapter ends by discussing why a gaze and gesture based interaction method is a feasible alternative to existing HCI techniques.

2.2 Interacting with the computer

Before discussing the failings of existing HCI techniques, this chapter first provides a brief overview of the commonly used techniques. Some have been used for many years and continue to be used, some are relatively more recent and some are still evolving.

Dix et al. (2004) identified the following key methods of interacting with computers:

- *Command line interface.* The user expresses the commands to the computer directly, using a set of predefined commands and function keys. Familiar examples of command line interfaces include MS-DOS (Microsoft Corp.) and UNIX environments.
- *Menus.* In menu-driven interfaces the user interacts with the system through a set of options displayed on the screen. The interaction is done either with the use of a mouse or through keystrokes, using function keys, numeric or combinations of function keys and

alphanumeric keys. The MS-DOS Editor (Microsoft Corp.) is an example of such an interface.

- *Pseudo Natural language*. These are interfaces where speech or written input in a natural form is used as a means of interaction. A few search engines, such as Ask Jeeves (Ask Jeeves), are examples of systems that use "natural language" processing.
- *Question/answer and query dialog*. These interfaces are based on the principle of asking the user a series of questions, to lead him/her through the interaction. An example of such an interface is an on-line air ticket booking system (British Airways).
- *Form-fills and spreadsheets*. *Form-fills* are very specific interfaces for data collection and are used both for data entry and for specifying fields in data retrieval. Interfaces that are used to enter data into large databases are usually of this type. Conference registration is an example of where such an interface is used (ACM SIGCHI). A *spreadsheet* is a system of rows and columns that forms a grid of cells in which the user may enter numeric or text data. An example of a widely used spreadsheet is Excel (Microsoft Corp.).
- *The WIMP interface*. Interfaces that use the combination of *W*indows – *I*cons – *M*enus – *P*ointers (WIMP) are the basis for most interactive systems used today. Examples of such systems are MacOS (Apple Computers) , Microsoft Windows (Microsoft Corp.) and Adobe Photoshop (Adobe)
- *Point-and-click interfaces*. Interfaces of this type are mainly used in multimedia applications and the web, as they enable the user to interact with the interface through a single click. The British Airways Check-in Kiosks (British Airways) found at most UK airports are an example of this type of interface
- *Three-dimensional interfaces*. A wide range of interfaces, starting from WIMP elements that have a sculptured effect to 3D virtual environments, belong to this category. Examples of 3D interfaces include Dive (SICS), COVEN (COVEN Consortium), VR-VIBE (Benford et al., 95) and UCL's ReaCToR CAVE environment (UCL).

Appendix A provides a more detailed description of each of these methods, with key examples given, as well as an analysis of their advantages and shortcomings

2.3 Problems with existing interaction techniques

The interaction techniques identified in the previous section certainly do provide benefits for Human Computer Interaction, as their continued use illustrates. However these techniques do raise certain issues, namely:

- **Existing interaction techniques are not natural and they typically require a user to learn a new interaction method**

  Most of the interaction techniques used today in HCI require the user to learn a specific interaction method for every type of system he/she needs to use, for example using command line interfaces and WIMP interfaces. These methods are typically significantly different from those that humans are more familiar with, namely human-human interaction techniques, which humans experience from birth onwards. As a consequence, humans have to invest time in learning and adapting to these alternative HCI interfaces. The fact that many interaction techniques heavily rely on memorisation of commands and sequences of operations can further increase the learning curve.

  Some interaction techniques try to overcome the difficulty imposed by memorisation by using logical groupings and hierarchies. This approach is illustrated by Menu and WIMP interfaces. The problem here is that a logical grouping for one person or group of people may not be accepted as a logical grouping by the larger group of users of a system.

  A similar issue arises with the use of visual metaphors (for example, button icons) to represent groupings or operations. It is not uncommon for a user to fail to recognise visual metaphors, and thus be forced to learn the mapping between the icon and its functionality.

- **Existing interaction techniques can be inconsistent**

  An issue that arises when users interact with a variety of interfaces that are of the same type is inconsistency. This can be very confusing for users, even if they are experienced. Typically users who use two similar interfaces that have some small differences can become confused and attempt to interact with one in ways they would with the other. A very familiar example is that of the trashcan icon in MacOS (Apple Computers) and MS-WINDOWS (Microsoft Corp.). In the Apple interfaces, placing a diskette in the trashcan ejects the diskette from the computer's diskette drive. In MS-Windows the same operation results in deleting the entire contents of the diskette.

- **Existing interaction techniques can restrict creativity and be over simplistic**

  Existing interaction techniques tend to be very structured and have specific purposes. Although this can assist in guiding a user through a particular process, this way of interacting can be seen as stifling creativity when it comes to interacting with computers. As the user has to perform specific tasks in pre-described ways there are very few opportunities to be creative. For example, a user who wants to create a story using a variety of media has to master a number of interfaces and finally edit together all the media clips using a different application. The user's creativity could be less stifled by the use of a single more creative interface, as the user would not have to become an expert in the use of a range of interfaces in order to be able to accomplish the creative task. The majority of creative interactions are in interfaces that have been specifically developed to support creativity, or are themselves a piece of digital artwork.

A related issue is the tendency to oversimplify interfaces. Although this is mainly a consequence of attempting to reduce the learning overhead and to gear an interface to a specific task, it has resulted in the development of interfaces that are very restrictive and specific. There is often no flexibility embedded in such interfaces and the interaction is simple but rigid. Consequently, should a new mode of working be required at a later date, it is likely that yet another interface would need to be developed. An example of such a case of interaction is a PDA based restaurant order taking system, where a point-and-click interface is used. If a customer wants to customise their meal, such as specifying no cheese in their salad, it will probably be up to the waiter to remember to inform the kitchen, as there is possibly no embedded flexibility in the system.

These issues suggest the need to consider alternative interaction methods. This thesis argues that such an alternative could involve making greater use of interaction techniques that humans use to communicate with each other, for example the use of gestures and gaze. The following section provides an overview of the key aspects of human-human interaction and discusses their applicability to HCI.

2.4 Aspects of Human-Human communication

This section identifies the main aspects of Human-Human interaction. It briefly examines three of these before the chapter focuses on gestures and gaze, the aspects that are primarily used in the work presented in this thesis.

Human-human interaction can be broken down into the following aspects (Dix et. al. 2004):

- Transfer effects and personal space
- Back channels, confirmation and interruption
- Turn-taking

- Eye contact and gaze
- Gestures and body language
- Speech

The following sections discuss each of these in more detail:

2.4.1 Transfer effects and personal space

When two people have a discussion they tend to stand with their heads and bodies at a fairly constant distance (Kendon, 1992). This is known as *personal space*, and refers to the distance from one person at which another feels comfortable. This distance is usually determined by the context in which the communication takes place. However, even in crowded or noisy areas, though the conversants will bring their heads together to "listen", they will then retreat back to their original position to reclaim their personal space. A typical situation where this is encountered is in elevators, where passengers will try to keep their personal space by any means, and as the elevator becomes more crowded the passengers tend to look away from each other by staring either on the floor numbers or their shoes so as to avoid eye contact and defend their personal space. Personal space is different across cultures, Britons preferring larger distances, North Americans and southern Europeans preferring closer distances.

When technology is used to facilitate human-human interaction, such as in video conferencing, transfer effect problems can occur as a result of the miscommunication of personal space. For example, during a videoconference, if one of the participants appears too zoomed in the other party might perceive this as intrusive and feel uncomfortable. In essence a virtual personal space is created which cannot be controlled by the participants (unless the zooming of cameras can be adjusted remotely). When supporting human-human communication with technology it is important to consider such implications.

2.4.2 Back channels, confirmation and interruption

When two people have a conversation, besides the actual dialogue there are a number of other communicative events or actions occurring, such as gestures, sounds, nods, grimaces and shrugs of the shoulders. All these activities define the tone and are appropriate to the context of the communication. They are of great significance, and their absence or their ignorance can affect the outcome of the communication. These activities are called back channels of communication (Dix et. al., 2004), and are compared in effectiveness with actual confirmation or interruption of an interaction. In their own way, they give feedback to the interactants. An example showing the importance of the back channels is a business meeting situation, where an idea is presented, and when feedback on the idea is given the person talking will appear to be really supportive or against the idea from the tone in his/her voice and the facial gestures he/she will make.

2.4.3 Turn-taking

One of the fundamental structures of a *dialogue* (a conversation between two or more persons, or similarly an exchange between a person and something else such as a computer) is turn-taking. When a dialogue takes place there is an informal protocol by which the parties involved wait until the speaker has concluded, at which point the next speaker in turn speaks, and the previous becomes a listener. This change of role, from speaker to listener and vice versa is called turn-taking (Dix et. al., 2004).

Back channels, discussed previously, help this process, by giving cues that a listener would like to be a speaker or that the speaker has exceeded the time that the listener is willing to listen for.

2.4.4 Discussion

Personal space, back channels and turn taking are very important aspects of human-human communication since they assist in defining the framework /context in which interaction takes place. When considering techniques to support humans interacting with computers it would be beneficial to bear in mind these characteristics, as they are already familiar to humans and are expected within an interaction.

The aspects that play a more fundamental role in human-human interaction are eye contact /gaze, gesture/body language and speech. These aspects actually carry the information that is communicated, whilst the aspects previously discussed mainly assist the communication. As discussed in Appendix A, the use of natural language as HCI technique is still problematic due to ambiguity. For this reason, this thesis focuses on the use of gaze and gestures as HCI techniques.

The following sections discuss these two aspects of human-human interaction in more depth and examine how they can be incorporated into HCI.

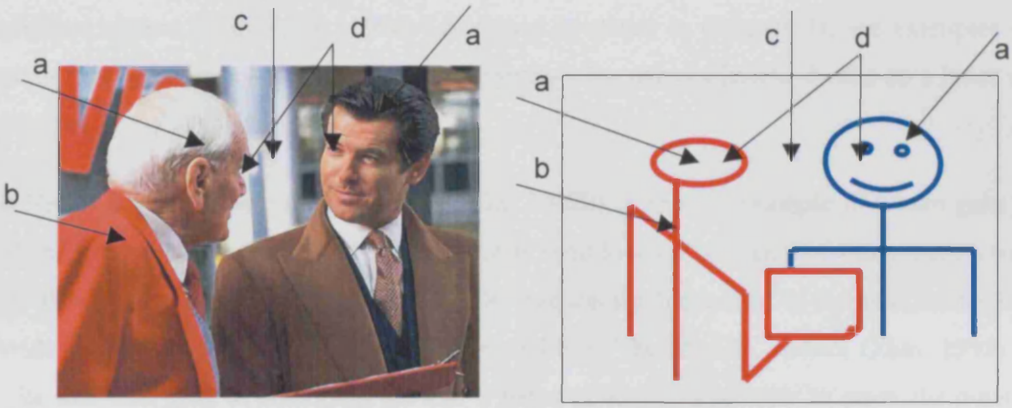
2.5 Gaze and Eye Contact

Eye gaze can be regarded as the instantaneous point of regard in the visual environment (Karn, 2002) and is both a communication channel and a body language signal for the recipient. *Eye* *contact*, which represents the percentage of time that two people who are interacting look at each other in the area of the face (Argyle, 1996), is used extensively in conversation. Through these channels we can identify issues such as whether our audience is listening or not, is bored, interested, confused etc. By directing our eye gaze and establishing eye contact with another party we can direct the communication towards that person or give a cue for that person to start speaking, i.e. a teacher may indicate a readiness to take questions from students by making eye contact with them (Kendon, 1992).

There are a number of different aspects of gaze which have different causes and effects on human-human communication (Argyle, 1996):

- *The Amount of Gaze at another.* This is the percentage of time spent looking at another person in the area of the face. People do not usually fixate at one area but make a series of glances at different points, usually around the eyes and the mouth.
- *Mutual gaze, or Eye Contact*, is the percentage of time that two people who are interacting look at each other in the area of the face.
- *Looking while talking and Looking while Listening*, Looking at the other participant, during a dialogue.
- *Glances.* Gaze consists of glances that have a duration of two to three seconds.
- *Mutual Glances.* As for Mutual Gaze, this is the percentage of time that two people who are interacting glance at each other in the area of the face at the same time. This is typically around one second.
- *Pattern of fixation*: the different points that a person fixates on when looking at another human or an object.
- *Pupil dilation.* This is an aspect of gaze that affects the behaviour of the recipient, though the recipient might not be aware of it.
- *Eye expression*: Eyes are expressive in many ways, such as how far open they are, the amount of white showing above and below the pupil. The recipient may characterise the gaze he/she receives based on eye expression as, for example, "staring", looking "intently" or "looking through" the other person, i.e. fixating beyond them (Fehr and Exline, 1987)
- *Direction of gaze-breaking*: when interactants are not looking at each other, and typically shift their gaze to the side
- *Blink rate*: This is an aspect that varies for a number of reasons such as anxiety, concentration and room temperature.

When two humans communicate, depending on their cultural background, they most likely will establish eye contact. Studies of face-to-face human–human interaction indicate that in such communication, typically both speaker and direct recipient orient their bodies at least partially toward one another. This is done in such a way that the orientation of each participant's head is toward the other's (Figure 2.1). This positioning enables the intermittent aiming of the eyes at another, which is one of the principle ways that the speaker indicates the focus of his or her attention (an experiment that illustrates this is described in chapter 3). In the same way, the direction of the recipient's eyes towards the speaker indicates an acknowledgment that he or she is the recipient (Kendon, 1992). So gaze can be described as both a signal and a communication channel in human interaction (Argyle, 1996).



(a) head turned towards other person, (b) body oriented towards the direction of other person,
(c) shared space (d) eye contact

Figure 2. 1- Two people engaged in a conversation (picture from MGM film "Tomorrow Never Dies" 1998)

2.5.1 Using Gaze as part of HCI

In HCI, the main benefit of gaze is to determine the focus of the user's attention, or, more specifically, determine what the user is looking at on the screen. The measurement of a user's attention can be used, for example, to help evaluate an interface, or to provide another input channel with which to control an interface.

One area in which research has focused is in the development of systems that use gaze to support communication. The GAZE groupware system (Vertegaal, 1999), is a virtual meeting room which uses gaze awareness in order to supplement audio conferencing, allowing each participant to see what the others are looking at, whether another participant or a document.

FRED (Vertegaal, 2001) is a multi agent conversational system that uses gaze to determine which agent the user is listening or speaking to.

Although gaze may seem to be most suitable for *supporting* communication it can also be used as a form of simple interaction. Instance et al. (1994) developed a system to evaluate how users can control unmodified graphical user interfaces, such as WIMP interfaces, through eye tracking. These experiment showed that simple tasks such as object selection and "clicking" can be satisfactorily achieved.

A simple interaction method has been to use gaze as a means to point-and-click by measuring how long the user's gaze dwells on an object on the screen (such a technique is referred to as 'look–and-dwell'). Work developed by Sibert et al. (2000), Salvucci (1999) and the EagleEyes system (Gips et al., 1996) (discussed in detail in Chapter 3), are examples of systems that make use of look-and-dwell for typing – the user looks and dwells on a letter on the screen.

The Visual mouse System (Farid et al., 2002a, 2002b) is another example of where gaze is used for interaction. In this case the users gaze is used to select a stream of video from a web page. Patmore et al. (1998) developed an EOG eye tracker (described in the next section) to provide a pointing device for people with disabilities. The MAGIC system (Zhai, 1999) is similar and uses gaze to determine the user's focus of attention in order to move the pointer around the screen (this system is discussed in more detail in chapter 3). Related to this is the system designed for WIMP environments by Lankford (2000), in which when a user dwells on an object, a menu with possible mouse operations appears.

Other systems includes that by Tanriverdi et al. (2000) that used an eye tracker for interacting with simple virtual environments in which objects move closer or further away, depending on whether the user is looking at them.

Generally, such systems as those above have demonstrated that such gaze based interaction can be quite successful for performing simple tasks. For more complex tasks, such as text selection and manipulation, the use of gaze can be more problematic (Instance et al., 1994). Other issues that these systems have encountered are the Midas Touch problem (Jacob, 1991) and equipment calibration in conjunction with user fatigue (Farid et al., 2002).

The research in this area has resulted in the development of numerous techniques to measure a user's gaze as well as identifying issues that need to be considered, in particular the so called *Midas Touch* problem. These are discussed in more detail in the following sections.

2.5.2 Techniques for Gaze based HCI

Eye tracking (monitoring where the eye fixates) is the most widely used method to determine a person's eye gaze. A variety of techniques have been developed to track a user's gaze, and the intrusiveness and restrictiveness of these may vary. The following list shows the more common eye tracking techniques that are currently used and these are classified according to the way they make contact with the user (Glenstrup, 1995):

- *Measuring the reflection of a light beam shone into the user's eye.*
- *Measuring the electrical potential around the eyes*
- *Using special contact lens that enable the tracking of the eye/lens position*

Besides eye tracking, alternative methods have also been developed that can be used to identify the focus of a user's gaze. In particular, studies have shown that the object of a person's gaze or focus of attention can be determined by their head orientation (Stiefalhagen and Zhu, 2002). This study is discussed in more detail in Chapter 3.

The above-mentioned eye tracking techniques are summarised here and compared for their usability and effectiveness. A more detailed description of these techniques is provided in Appendix B.

2.5.2.1 Techniques based on Reflected Light

These techniques are based on the reflection of light (normally infrared) on some part of the eye. Infrared light is preferred in eye tracking because it is "invisible" to the eye. This means that it does not distract the user, and, as infrared detectors are generally not affected by other light sources, there are no special lighting requirements. The *Limbus* and *Pupil Tracking* methods (Glenstrup, 1995), otherwise referred to as *Infrared Oculography (IROG)* (Eye-movement equipment database), are based on the *amount* of light that is reflected back from the eye. Other methods are based on the Perkinje (Muller et al., 1993) phenomenon (as described below in section 2.5.2.1.1), and use specific reflections that occur on the boundaries of the lens and the cornea. The most popular eye tracking method, the *Corneal and Pupil reflection relationship – Video Oculography (VOG)* (Eye-movement equipment database), uses this approach (and is reviewed in the following section).

Problems that arise in association with these methods include the amount by which the eyelids cover the user's eyes, the contrast between the pupil and the iris, and that in all methods the measurements are always in relation to the head. This has as an effect that either the user's head has to be restricted in movement (this is achieved in many cases by placing the user's head in a frame) or that the user wears the equipment on his/her head. Clearly, both of these methods can be intrusive to the user.

The range of applications in which these eye-tracking techniques are used extends beyond HCI. Ergonomics, neurology, ophalmology, sleep disorders, and aviation are areas where this technique is used both in research and in specialised applications. The majority of eye tracking systems today use reflected light techniques. VOG is considered more comfortable, and less intrusive, and for this reason is used also with children. This technique is discussed in more detail below.

2.5.2.1.1 Corneal and Pupil reflection relationship – Video Oculography (VOG)

When light, typically infrared light, is shone into the user's eye, four reflections occur on the boundaries of the lens and the cornea. This is due to the Perkinje phenomenon in which all of the colours of the spectrum do not fade equally with diminishing light. It is actually a shift in the relative brightness of certain colours as illumination diminishes. These images are referred to as the Perkinje images (figure 2.2). The first Perkinje image, or *glint* as it is also called, together with the reflection off the retina, or bright-eye as it is also referred to, can be recorded using an infrared sensitive camera. As the user's eye moves horizontally or vertically, the relative positioning of these two images change accordingly. The direction of the user's gaze can be determined by calculating from these relative positions and this is referred to as video oculography (VOG).

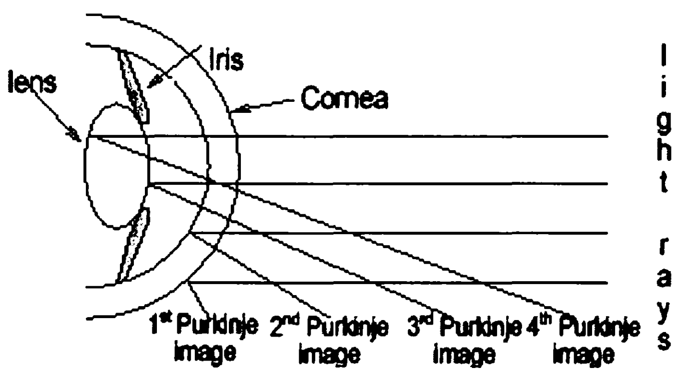


Figure 2. 2 - Purkinje images (Glenstrup et al., 1995)

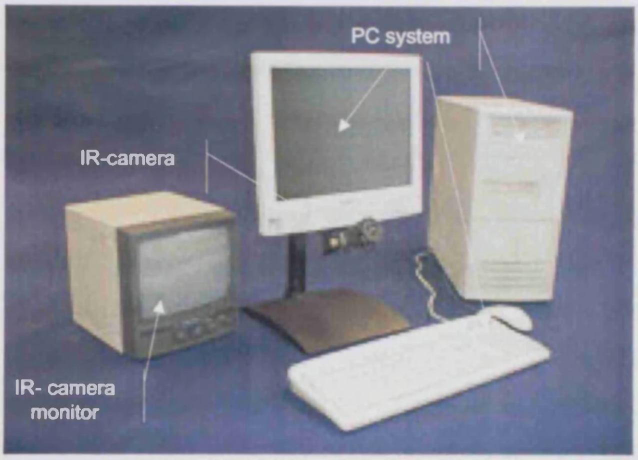


Figure 2. 3 - LC technologies eye tracker

The main problem with this technique is that a good view of the eye must be maintained, so the head movements must again be minimal. For this reason a VOG based system is either worn by the user on his/her head or may require a movement-restricting frame or chin rest. A typical eye tracking system, where the IR-camera is mounted under the monitor, is shown in Figure 2.3.

Numerous eye tracking systems have been developed that make use of the VOG technique, including the Visual Mouse system (Farid el al, 2002), the Gazetracker System (Lankford, 2000), the FRED system (Vertegaal, 2001), The Gaze groupware system (Vertegaal, 1999), and other systems (Sibert et al., 2000; Tanriverdi et al., 2000; Salvucci, 1999 and Instance et al, 1999).

2.5.2.2 Technique based on the electric potential around the eyes

This technique is based on electrooculography, or EOG, which is a direct recording of the electrical potentials generated by eye movement. In saccadic movement of the eye (side to side eye movement, for example that occurs when reading), when the eye focuses on a particular spot the potential remains constant. Depending on the direction of the saccadic movement of the eye, the potential will be either positive values (for left movement) or negative values (for right movement), as shown in figure 2.4 (Hasset, 1978). EOG measured eye movements have been useful in the assessment of cognitive functions (Stern, 1984), drug effects (Jantti, 1983), path physiology (Kennard et. al, 1994) and psychiatric disorders (Holzman and Levy, 1977) and in the accurate assessment of eye fixation in the operation of an aircraft (Viveash et. al , 1996).

Tecce et al. conducted two experiments to determine the accuracy of computer control through eye movement using the EOG method (Tecce et al., 1998). Their findings concluded that the EOG method can permit successful control of computer functions (in this case eye typing). The main disadvantage of this technique is that the electrodes are positioned on the user's face, very close to the eyes, which is very intrusive.

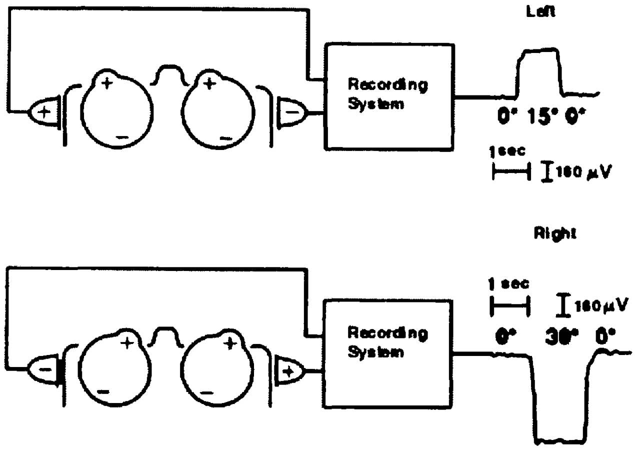


Figure 2. 4 : Measurement of electrical potential in saccadic movement of the eyes (Hasset, 1978)

The EOG method has been successfully used by Gips and Olivieri (1996) in the EagleEyes project. Patmore and Knapp (1998) have also used this technique successfully for computer control, in which the user can move the cursor to given target areas on the screen.

2.5.2.3 Techniques Based on Contact Lenses

It is also possible to make recordings of the direction of a human's eyes with the use of special contact lenses. Two methods are commonly used. In the first, the contact lenses have mirror surfaces engraved on them and the reflection of the light beams are used to calculate the position of the user's eye. In the second method, a tiny induction coil is implanted into the lens, and this allows the recording of the user's eye position through the use of a high-frequency electromagnetic field, placed around the user's head.

Both of these methods are problematic, as they are intrusive and may be very uncomfortable for the user, as some available systems have wires connected to the contact lenses. There are also health issues concerning high-frequency electro-magnetic fields. As a result, contact lens based eye tracking has not become widely used, with developers preferring to use techniques based on reflected light instead.

2.5.2.4 Summary of techniques

All of the above techniques possess many similarities but also have some very important differences. For example, VOG and EOG produce oculograms in different ways. In addition, the equipment used to gather the input data can vary in cost, complexity and how intrusive it is for the user. For example, the use of a camera mounted on a monitor is less intrusive compared to the positioning of electrodes around the user's eyes.

Of all of the above techniques, however, the most popular with developers are those that use reflected light, and in particular VOG.

2.5.3 The Midas Touch problem

Considering gaze based interfaces and the interaction options that are available with the implementation of the technologies discussed previously, it is very easy to design an interface that will present a similar problem to what Midas experienced. Jacob (1991) describes such a situation: "At first, it is empowering to be able simply to look at what you want and have it happen, rather than having to look at it (as you would anyway and then point-and-click it with the mouse or otherwise issue a command). Before long, though, it becomes like The Midas Touch. Everywhere you look, another command is activated; you cannot look anywhere without issuing a command. The challenge in building a useful eye tracker interface is to avoid The Midas Touch problem."

This problem is illustrated through a brief discussion of two versions of the "spell and speak" program that were developed in the EagleEyes project (discussed in more detail in chapter 3). The first puts a picture of a keyboard on the screen (figure 2.5). By dwelling his/her gaze on a letter the user selects that letter, and it is placed in the message area. In order to delete a selected letter the user may "select" the backspace. By selecting "end" when the user is finished with the letter selections the computer "speaks out" the message.

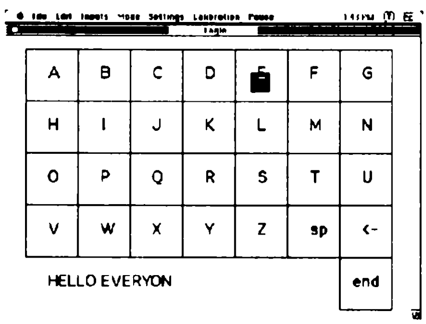


Figure 2. 5 - EagleEyes full screen "spell and speak " (Gips et al., 1996)

The second program is more suitable for people who are not familiar with a keyboard. The letters of the alphabet are grouped in groups of 5 or 6 letters (figure 2.6). The user selects the group of letters that he/she wants to use and these appear in the boxes at the lower part of the screen. The user may then select the letter needed and it is placed in the message area in the middle of the screen. This process continues until all the letters of the message are inserted.

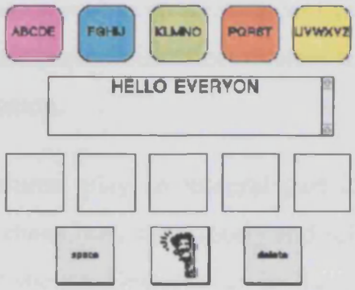


Figure 2. 6 - EagleEyes two-level "spell and speak" (Gips et al., 1996)

Considering these two applications of the same technique it is clear that the first can suffer acutely from the Midas touch problem. Wherever the user looks on the screen there is a key, which at a fixation-dwell of the eye will be selected. The user has to be careful as to where to rest his/her gaze. However, the second program provides additional "dead" space that can be used to look at when not intending to select a key (Gips and Oliviery, 1996). Consequently, the Midas touch problem is less of an issue with this version.

The Midas Touch problem is clearly an important issue for designers of gaze based interfaces.

2.5.4 Summary

Gaze can be an effective additional modality for interacting with computers. Research over the past years has resulted in the development of numerous systems that use gaze either as the only, or as a supplementary, modality to some effect. This research has also led into the development of a variety of techniques that enable us to capture the user's eye movements.

Though the use of gaze can benefit HCI, a number of limitations exist that render it problematic for certain tasks. In particular, gaze is less effective when used for achieving complicated tasks (for example, editing and manipulating an image), because humans naturally use gaze to observe rather than to control. The use of gaze also raises problems for interface design, in particular the Midas Touch problem. Furthermore, there are a number of problems associated with the hardware used to capture gaze such as calibration of the system, user fatigue and intrusiveness.

Despite these drawbacks, gaze can prove to be effective for simple HCI tasks.

2.6 Gestures

Bauml and Bauml (1997), in their *Dictionary of Worldwide Gestures*, define a Gesture as: "a posture or movement of the body or any of its members, that is to be understood to be meaningful". Gestures themselves, may or may not accompany speech but are as ephemeral as the spoken word. They are either a visible accompaniment to or a substitute for, speech or action. Whenever a gesture by one person influences another it is referred to as an instance of non–verbal or bodily communication.

Gestures, particularly hand gestures, play an integral part in our everyday human-human communication. Humans use gestures both consciously and subconsciously and they are used either on their own or to support speech. Gestures can be the main communication channel or a back channel (as highlighted in section 2.4).

The context in which people use gestures can vary and includes:

- *An accompaniment to normal speech.* For example, when a person is describing an object, he/she may often show the dimensions of that object by gesturing.

- *A substitute for a foreign language.* For example, communication through signs was extensively practised by the Plains Indians to overcome the variety of languages and dialects among their nations.

- *A substitute where normal speech becomes inaudible, disadvantageous or dangerous.* For example, the gestures used by divers to communicate under water.

- *An accompaniment to certain professional activities.* For example, by actors, dancers and practical speakers, to supplement or replace the spoken word.

Gestures have been classified according to the way they are integrated in our everyday communication and the way they are formulated. McNeil (1992) classifies gestures in five categories, according to their use and way of delivery:

- *Iconics*: These are gestures that bear close formal relationship to the semantic content of the speech. For example, when a person is describing a fight, he appears to replicate with his fists the movements of the fighter.

- *Metaphorics*: These gestures are similar to iconic gestures in that they are pictorial, but in this case the pictorial content presents an abstract idea instead of a concrete object or event. For example, when a narrator is describing a scene from a Tom and Jerry cartoon, he/she might define a space by placing his/her hands facing each other at some distance apart. This gesture defines the setting or the proximity of the characters but not any of the actual action of the scene.

- *Beats:* These gestures are named so because they look like beating music time. The hand moves along with the rhythm of speech. They reveal the speaker's conception of the narrative dialogue as a total. The semiotic value of a beat emphasises the word or phrase it accompanies, not for its semantic value, but at the specific point that it occurs in a dialogue. A typical example of a beat gesture is found during discussion, when a speaker wants to emphasize a point such as an event that happens frequently, and does this by slightly raising his/ her hand, and then bringing it back to a resting position.

- *Cohesive:* The cohesive gestures are quite eclectic in form. They can consist of iconics, metaphorics, pointing gestures, or even beats. Gestural cohesion depends on the repetition of the same gesture form, movement, or locus (i.e. the location of the gesture) in the gesture space. Through repetition, the gesture shows the recurrence or continuation of the theme. Examples of cohesive beats are the gestures that are demonstrated by politicians whilst talking.

- *Deictic (pointing):* Pointing has the obvious function of indicating objects and events in the concrete world, but it also plays a part even where there is nothing objectively present to point at. Most pointing gestures in narratives and conversations are of this abstract kind. For example, during a conversation a speaker might ask his /her conversants about their whereabouts, pointing to an area between them, although the discussion might be referring to a different time and location.

2.6.1 Gestures and language

Gestures are closely linked to speech, but present meaning in a different form. Our thoughts are expressed by a combination of gestures and utterances. Thoughts may be comprised of imagery, specific concepts or generalisations, which can be subdivided into smaller segments. (McNeill, 1992). To express our thoughts we typically use both speech and gesture.

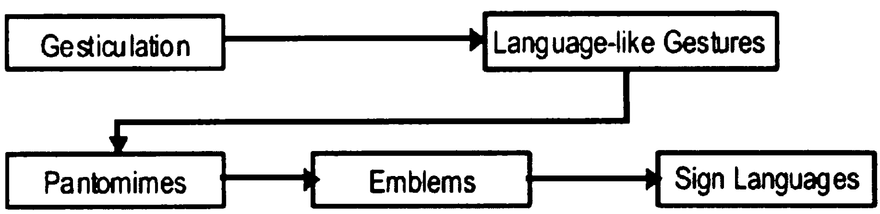


Figure 2. 7 - Continuum of Gestures (McNeil, 1992)

Kendon developed a "Continuum of Gestures" (McNeil, 1992) to show the process of articulating a word or concept through a gesture (figure 2.7). Kendon's continuum is important as it can be used for distinguishing gestures of fundamentally different kinds. McNeill uses the term "gesture" specifically to refer to the leftmost *gesticulation* end of his spectrum.

In the *gesticulation* sense, gestures are idiosyncratic movements of the hands and arms accompanying speech. These almost never occur in the absence of speech

Language-like gestures are similar in form but are grammatically integrated into the utterance

In *pantomime*, the hands represent objects or actions, but speech is not obligatory. The fading of speech brings pantomime in the middle of Kendon's continuum. In a pantomime there can be either silence or inarticulate onomatopoetic sound effects, like "ssh!", "Click", etc. Successive pantomimes can create sequence-like demonstrations and this is different from gesticulation where successive gestures do not combine.

Emblems are the "Italianate" gestures, mostly insults, though in some cases praise. Virtually all emblems represent an attempt to exert control over another person. Emblems have standards of well-formedness, a crucial language-like property that gesticulation and pantomime lack.

Sign languages are full-fledged linguistic systems.

When considering the use of gestures in HCI, it is useful to consider this categorisation of gestures that we use in our everyday lives. The gestures that are at the beginning of the spectrum are more natural and idiosyncratic, while those at the end are more standardised. According to the type of application, there are cases that a specific type of gesture is more appropriate that another. The majority of applications use emblem type gestures.

Gestures are discussed in greater detail in Appendix C.

2.6.2 Using Gestures in HCI

For HCI, gestures can provide an alternative interaction technique. Users may then perform tasks in a way that appears more natural, and closer to the way they carry out similar tasks in their everyday life. For example a PDA user may use a stylus to write in a similar way that he/ she would write using a pen. The difference, however, is that the gestures performed with the stylus may not necessarily represent the characters in the alphabet (for example, the letter 'A' could be represented by the gesture 'Λ'). The effectiveness of the use of gestures in HCI depends on the use of the appropriate technique (for example a mouse is appropriate for interacting in a WIMP environment, but when coming to freehand drawing it is not as suitable as a light pen). The use of gestures within HCI is largely focused into two main types, 2D and 3D gestures.

Since the invention of the mouse and the development of GUIs, two-dimensional gestures have been integrated in everyday HCI. Additionally, the development of PDAs and mobile phones with small touch sensitive displays has seen styluses been used as a gestural input device (Sony Ericsson). A very familiar example is the use of a PDA where a stylus is used for note taking during a meeting or a conference session, for example on an iPAQ (HP Invent). The use of a light pen by a graphics artist or an engineer to produce a design (with packages such as AutoCad), or the use of mouse generated gestures to create a musical score (Rubine, 1992), are other examples of the application of 2D gestures. 2D Gestures are also used in gaming, for example in the game Black & White (Electronic Arts) the user may cast a spell by performing one of 20 gestures (such as drawing a "W") with the mouse by holding down the left mouse button. The game identifies the gesture and responds accordingly.

The use of 2D gestures has been very successful and they have been integrated in our everyday use of computers. This is due to the fact that most of the gestures that are performed in this 2D manner are performed in the same way in their usual context (for example drawing). Problems do occur, however, especially when precision is required. This is because at that point the gesture is no longer natural, but has to respond to the constraints of the interface in order to achieve the desired outcome. Users have to spend time to learn to use devices such as the mouse and stylus with grater accuracy. This can be cumbersome and in some cases, prolonged use of such devices may also cause problems such as RSI (repetitive stress injury).

Three-Dimensional gestures are still not widely used, although a number of techniques and devices are available (as discussed in the following section). Most of these are research prototypes, or specific installations. The main benefit of 3D gesture based interfaces is the use of gestures that are performed naturally in the context of completing a specific task. In some systems the gesture interaction can be very simple, such as pointing, as used in the "put-that-there" system (Bolt, 1980). 3D gestures have also been used to control parts of a system, such as cameras in teleconferencing situations (Howell, 2002, and Herpers, 2001).

The use of gestures in music has also received much attention, most likely because gestures are commonly used by humans to both express and to perform music (for example in the use of musical instruments). The Digital Baton (Marrin, 1997) is a system that uses gestures to control and perform music. WorldBeat (Borchers, 1997) at ARS in Austria, uses a gesture interface to allow users to navigate and learn through the experience of interacting with a musical exhibit. Both of these systems are examined in more detail in chapter 3. Arfib et al. (2002) use gestures to synthesise music, while Willier et al. (2002) use the gestures produced by a juggler to control music. These systems tend to allow more sophisticated control via the use of gestures, whilst also providing the user with clear and immediate feedback (for example, changing the tempo, or playing the drum).

As technology has developed over the years, it has become possible to combine more modalities and to handle media in a more effective way. Bolt (1992) used a combination of gestures, voice, and eye tracking to manipulate on screen objects (for example, rotating a cube) with the use of data gloves. A similar idea is used in the SGIM project (Sowa et al., 1999) where gestures are one of the multimodal inputs used to manipulate objects in virtual construction (for example, constructing a simple car in a virtual reality system). Thorisson (1997) used gestures as one of the modalities in the multimodal dialog between a human and Gandalf, an embodied humanoid agent. The ongoing SmartKom project is developing an environment which will use gestures as an input modality and aims to provide the user with an interface that will accommodate most of the natural human senses (SmartKom Consortium).

Other developments have focused on the system learning to recognise the gestures. Triesch et al. (1999) have used gestures to train a robot to perform similar tasks, while systems have also been developed to recognise sign language (Bauer et al., 2002), or to provide tools for signers. An example is SignPS, which is an interactive printing system for sign languages (SignPS).

Generally, gestures have been successfully incorporated into HCI, with various studies of the developed prototypes showing that they support effective interaction. The spectrum of tasks varies from pointing to onscreen objects through to object manipulation and sign language recognition. The main drawback in the adoption of gestures in HCI is the technology involved. Although the cost of available hardware for use in 2D gesture capture is relatively low, the cost of technology to capture the more sophisticated 3D gestures, such as Data Gloves, are much more expensive (ranging from $500 to $14000). Such technologies are also obtrusive, often requiring to be worn by the user. The positioning of sensors and cables can make users feel restricted in their movements. However, these issues are being resolved, as recent technological developments of less intrusive and more affordable devices have made it more feasible to use 3D gestures in everyday computer interaction.

Besides the hardware related issues, attention must also be drawn to the type of gestures used and how they are interpreted. In certain cases, such as sign language, depending on the actual language being used (for example American sign Language and British sign language), a specific sign might have a completely different meaning. Other issues include how "natural" the gestures that users have to perform are, and how much training is required in order to effectively use a gesture-based system.

The work proposed in this thesis utilises 3D hand gestures as a means of interacting with a computer. The following section describes the key techniques used to capture hand gestures, with a focus on 3D capture techniques.

2.6.3 Techniques for Gesture based HCI

Depending on the type of gesture performed by the user, a variety of techniques can be used. As already discussed, the main discriminating factor is whether the gesture is 2D or 3D. In the brief analysis carried out here 2D Gestures have been examined as one category, while 3D gestures, which are used in the work presented in this thesis, are discussed in more detail. 2D gestures are now used by almost all computer users and are integrated into our every day human computer interactions. These techniques tend not to be as intrusive as those employed in 3D gesture recognition. The techniques described here are the most commonly found gesture capture techniques, and have been considered in the selection of equipment for use in the work presented in this thesis.

- *2-Dimensional Gesture capture techniques*
- *3-Dimensional Gesture capture techniques*
    - *techniques using physical devices*
    - *techniques using Data gloves*
    - *techniques using Computer Vision.*

2.6.3.1 Two-Dimensional Gesture capture techniques

Most computer users today use a pointing input device that allows them to perform 2D gestures that are used to interact with the computer. The most common device is a mouse that allows the users to perform a gesture on the surface of their desk and this is transposed as a movement of the cursor/arrow on the computer's display. Besides the mouse, devices such as trackballs, track-point buttons, mini joysticks, light pens, and touch pads have also been used as instruments for such 2D gestures. This type of 2D gesture has been successfully used within WIMP (MacOS), design (CAD, Computer Aided Design), games (Electronic Arts), and music editing (Rubine, 1992) interfaces, as well as embedded within hardware such as laptop computers (IBM ThinkPad, Sony Vaio), PDAs and mobile phones (Sony Ericsson).

These gesture capture devices typically provide a space in which gestures must be performed (for example, a mouse mat, touch pad, on a joystick, etc) that is usually separate from the actual work area (as perceived by the user). However, alternative devices, such as touch sensitive displays (iPAQ), have also been developed to enable users to perform more direct gestures. For example, users can perform typical WIMP tasks by pointing and dragging their finger on the display (as often found in information kiosks and internet phone booths).

A disadvantage of 2D gesture capturing techniques is that typically the gestures that are performed using them are deictic in nature, due to the 2D constraint. This limitation has caused the development of tools to reproduce the functionality of 3D gestures, for example, the selection marquee tool combined with the rotation tool (Adobe Photoshop). Although 3D gestures can be emulated to an extent, it is not possible for 2D gestures to possess the same functionality and flexibility that can be provided by 3D gestures.

A significant advantage of the devices used to perform 2D gestures, however, is their relatively low cost, a factor that makes them accessible to the average computer user. On the other hand the functionality is limited to the constraint of the 2D, as operations that would apply to a third dimension have to be simulated or split to a different view, as in 3DMAX (Discreet).

2.6.3.2 Three-Dimensional Gesture capture techniques using physical devices

Humans make use of physical devices in their everyday lives to perform a variety of tasks and normally in the process they inadvertently use these devices to perform gestures. In order to exploit these gestures in HCI a number of techniques and devices have been developed. Three common techniques are: Phicons (physical icons), haptic wands and batons.

Physical icons (or Phicons, Ishii et al., 1998) are techniques that can be used to capture hand gestures. The idea behind Phicons is that as there are objects that we use in our everyday life where particular actions that involve these objects can be mapped to particular events. For example, in the ambientRoom (Ishii et al., 1998), removing the cork from a small glass bottle results in a sound being produced that represents network traffic. Other systems that have used phicons are Tangible bits (Ishii et al., 1997) and Rasa (McGee et al., 2000). One advantage of Phicons is that the user can make the mapping between actions and events themselves and so in a sense the user can customise the interaction. The main drawback, however, is that only some human activities map onto interactions with physical objects and not all people use objects in the same way. For example, when indicating a direction, humans do not necessarily use an object to point to the specific direction, but somebody holding an umbrella might use it in this way.

Haptic wands are devices that enable the user to perform 3D gestures, and they provide force feedback. For example in a sculpturing simulation, the emulated scalpel could allow the user to feel the resistance of the material that is being sculpted. An example of a haptic wand is the Phantom (SensAble technologies). As shown in figure 2.8, it has a pen like wand that the user can move around in space or, if required, the user can substitute the wand with his/her finger. Applications that have used haptic wands range from collaborative virtual environments (Sallnas, 2000), allowing blind people to haptically read graphs (Yu et al., 2000), through to their use in assisting surgical training (Angus et al., 2002).

The main disadvantage of haptic wands is that the devices are quite restrictive in the extent that a gesture can be performed, although they do tend to work well for high precision gestures. However, because wands give haptic feedback the user can be aware of the space and objects that he/she manipulates. For these reasons the main application of such devices is in surgical training and assistive systems for blind people. The very high cost of the devices is also a disadvantage.

Figure 2. 8 - Phatom (SensAble technologies)

Finally, devices such as batons can be used to produce or enhance a gesture (as used in real life; for example, a teacher using a ruler to point on a map). As the user performs a sequence of gestures using the baton, sensors track the positions that the baton moves through. A computer then maps the sequence of positions to a gesture. Batons typically make use of either infrared, or a combination of infrared and orthogonal accelerometers, to capture a user's gesture. Example systems that use batons include WorldBeat (Borchers, 1997) and the digital baton (Marin, 1997). Both of these are discussed in more detail in chapter 3. Batons are most suitable for interfaces that mimic situations where the user would use a baton type object, such as a musical instrument or a stick pointer. Outside these situations they can be less practical, as well as forcing the user to utilise a potentially less natural interaction style. Batons also require a certain amount of space for their use, which can be a problem.

Physical devices can be effectively used to perform gestures associated with actions that a user would perform in a natural setting. When mapping gestures that are not performed in a natural setting, the use of such devices can be problematic (for example, the amount of space required may be prohibitive).

2.6.3.3 Three-Dimensional Gesture capture techniques using Data gloves

Data gloves are wearable devices, based on gloves that have sensors, which capture the movements made by the user's hands. There are mainly two technologies that are used in Data gloves: optical fibres and sensors.

Optical fibre based gloves

Optical fibre based data gloves (5DT), are typically made out of flexible material (commonly Lycra), with an optical fibre based sensor embedded along the length of each finger, as well as tilt sensors on the top of the hand (Figure 2.9). These sensors can typically measure the flexure of each finger and the roll and pitch angle of the hand, up to 200 times per second and communicate this back to a computer (both cable and wireless systems exist). Generally, their precision is good, making them suitable in computer gaming and interfaces that do not require high precision of hand movement identification. Gloves of this type are less successful at measuring delicate hand movements where high precision is required.

A further problem that may arise with this type of glove is that of glove size. Although they are modestly priced (typically around $500) they are still too expensive to make the production of numerous glove sizes practical. As a result the glove might be too large for the average female hand. In turn, this can affect the effectiveness of the glove as the fibre optical sensor loop around each finger assumes that the glove fits well - the flexure of the finger should cause the flexure of the sensor.

Figure 2. 9 - 5DT Data Glove

However, optical fibre based gloves are lightweight, easily configured and calibrated and have been successfully used within virtual environments (Grant, 1998; Lee, 2002).

Sensor based gloves

Sensor based data gloves (CyberGlove, SuperGlove) use isolated sensors that monitor the motions of the hand and its fingers. The number of sensors may vary, but typically there are 18 and they are placed over the joints of the hand and wrist (figure 2.10). There are two sensors on the Thumb that measure the metacarpophalangeal (MP) joint and the interpahlangeal (IP) joint. There are two sensors on each of the four fingers that are also placed on the MP and the proximal IP joints (marked 1 and 2 on figure 2.10). Additionally, sensors that measure the amount each finger moves laterally are placed on the topside of the glove for each of the five fingers. The thumb and the little finger also possess sensors that measure the rotation of these fingers across the palm towards each other. The remaining two sensors are on the wrist and measure the pitch and the flexion of the wrist. On 22 sensor gloves an additional sensor is added on the distal IP joint of every finger (3 on figure 2.10).

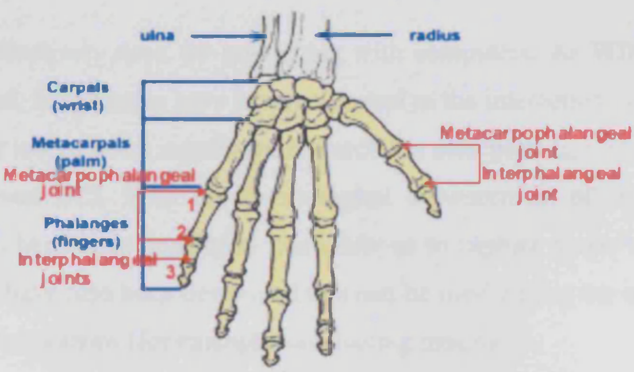


Figure 2. 10 - Right Hand

The main disadvantage of this type of glove is its cost (around $10,000-$15,000, depending on the number of sensors). The cost makes it unaffordable for general use and even for research, in many cases. However, this type of glove has very high precision and is ideal for interfaces that require this (such as simulations of medical operations or sensitive equipment). This type of glove is only slightly affected by the glove size problem highlighted above due to the position of the sensors. The large number of sensors, however, does mean that they are cable bound and this may restrict the movements of the user. Despite this, these gloves have been successfully used by Bolt and Herranz (1992) and by Thorisson (1996).

2.6.3.4 Three-Dimensional gesture based systems using Computer Vision.

A large number of systems that use gestures, mainly in experimental stages, use computer vision for capturing hand gesture. This technique is based on the analysis of the information captured by a camera, i.e. the analysis of an image of the hand. A variety of techniques and algorithms are used in this analysis. Some examine static hand posture (Laptev, 2001), others identify fingertips (for example, the GREFIT system of Nokler, 1999), and others identify blobs of skin colour (Marcel, 1999), or even coloured gloves. The SmartKom project (Beringer 2001), SignPS and work by Triesch et al. (1999), have all used this technique to capture gestures and it has also been used successfully in the film industry to capture whole body movements in order to reproduce them through animation (e.g., Lord of the Rings, 2002).

The main advantage of the computer vision technique is that there is no need for direct contact with an interface, so the user does not have to be conscious of the interaction at all times. The main disadvantage of this technique is the fact that the camera(s) used are typically permanently mounted in a room, which restricts the use of such a system to a specific location.

2.6.4 Summary

Gestures can be effectively used for interacting with computers. As WIMP interfaces have become widely used, 2D gestures have become central to the interaction between the user and the computer. Over recent years significant research has also been carried out within the area of 3D gesture driven HCI. With the technological advancement of computer vision and sensors, techniques have been developed that allow us to capture a user's hand movements. Similar techniques have also been developed that can be used during the specific activities in which gestures are performed (for example, conducting music).

The use of gestures within HCI has had a significant impact in the way people use computers today. However, when it comes to 3D gestures there are still a number of problems that have to be resolved. Technological advances have made it possible to develop a number of devices, but their cost is still high. Additionally, devices such as Data Gloves have the problems that occur with all wearable devices, i.e. they can be intrusive, they do not always fit, they might not be comfortable and the cables connecting to the computer are restrictive. There are also issues that have to be considered when designing the interfaces that will use gestures, including defining the natural locus of a gesture, whether the gesture is 3D or if it is naturally 2D, whether the gesture requires the user to have a large personal space, or whether it can be confined to the area between the user and the screen.

Although there are issues still to be considered, gestures can play a central role within human computer interaction.

2.7 Discussion

This thesis argues that the use of eye gaze and 3D hand gestures is a feasible and beneficial means of interacting with computers. The proposed Conductor Interaction Method aims to be more natural and intuitive to use by a user. As has been previously discussed, the interpretation of natural language is still difficult for computers, and for this reason this thesis focuses on gaze and hand gestures as a means for interaction rather than speech.

The above argument for a gaze and 3D hand gesture interaction method is supported by a number of factors:

- ***Improved interaction***

  As discussed in section 2.3, existing Human-Computer interaction techniques (such as WIMP interfaces) suffer because users have to learn new, typically unnatural, interaction methods. These interfaces also tend to possess inconsistencies whilst also restricting user creativity. Making use of Human-Human interaction methods, such as gaze and hand gestures, can overcome some of these issues and have the advantage that they are typically interaction styles that users have learned to become familiar with from a very young age .

  As a consequence it can be argued that the adoption of Human-Human interaction methods within Human-Computer interaction can result in improved interfaces and interaction styles. This is likely to be particularly true for novice computer users. Furthermore, for certain tasks the application of these modalities can enable the use of more appropriate tools.

- ***Technological advances and reduced costs***

  Technological advances and reduced costs have made it more feasible for Human-Human interaction methods to be supported within HCI. With the development of low cost devices such as mice and touch-pads, as well as the affordability of more sophisticated devices such as optical fibre data gloves, Human-Human interaction methods can now be more realistically considered.

  Despite the above advances, however, there remain issues such as intrusiveness, precision, and, in some cases, high costs to be considered. Intrusiveness and user fatigue are probably the most important usability issues associated with these methods. Precision is important for the type of applications that can adopt these interaction techniques, and the cost is a determinant of the rate that these technologies will be adopted by larger numbers of users.

However, the technologies are now at a point where it is no longer infeasible to consider the use of Human-Human interaction methods within HCI.

- ***A greater need to explore the possibilities of integrating Human-Human interaction methods within HCI***

Given the technological advances and reduced costs of devices that can support Human-Human interaction techniques within HCI, the shortcomings that are possessed by existing interaction styles and the ever increasing number of computer users (and in particular the increasing number of non-expert users), it is highly likely that there will be a growing trend towards integrating aspects of Human-Human interaction methods within HCI. Consequently, there is a growing need for research into new techniques and interfaces that can cater for this.

The Conductor Interaction Method, which incorporates the gaze and gesture modalities, and the architecture for the proposed Presentation Conductor system that realises this method, are described in chapter 4. An implementation that realises this architecture is presented in chapter 5. Chapter 6 provides an evaluation of the proposed method and the implementation that utilises it.

2.8 Chapter summary

This chapter has provided an overview of eight common HCI techniques, and has highlighted their failings with respect to providing a natural interface for interaction. In particular, these techniques are not inherently natural, and typically require a user to learn a new interaction method. These interaction techniques can also be inconsistent as well as being over simplistic and stifling user creativity.

This chapter has argued that, to a large extent, these failings can be overcome by drawing from interaction techniques used in Human-Human interaction. More specifically, the use of eye gaze and 3D hand gestures has been examined. An overview of the most commonly used techniques for capturing these modalities is provided and it has been argued that these modalities represent a suitable means for interaction in the novel interaction method that is proposed by this thesis. This argument is further supported by the recent advances and reduced cost in the relevant technologies.

The next chapter examines nine key systems that are relevant to the work proposed in this thesis. The analysis that is performed identifies experiences and the lessons to be learnt and that informed the design of the Conductor Interaction Method proposed in this thesis.

Chapter 3 – Related Work

3.1 Introduction

This thesis presents the development of a novel interaction scheme, where the user interacts with a computer in a more natural way than with existing techniques. In chapter 2 it was argued how existing commonly used HCI techniques fail to provide users with a natural interface for interaction. In particular, it was pointed out that these techniques typically require the user to learn a new interaction method, tend to be over simplistic, and stifle user creativity. It was argued that a more suitable alternative might be to draw on Human-Human interaction techniques, and in particular gaze and gestures.

This chapter examines nine key systems that use gaze, gestures or a combination of these as a means of interaction. This analysis can identify the experiences and the lessons that can be learnt from these systems, which, in turn, help inform the design of the interaction method presented by this thesis. In particular some of the systems discussed use a two-phased interaction method, combining both the gaze and gesture modalities, which is related to the work presented in this thesis.

To help structure the chapter, the systems are presented and discussed within the following categories:

- Gesture only systems
- Gaze and head tracking systems
- Multiple modality systems

3.2 Gesture only systems

This section discusses three systems that make use of gestures as a mechanism for interaction.

3.2.1 Gscore

The Gscore editor (Rubine, 1992), developed at Carnegie Mellon University, enables users to use gestures to place and manipulate notes on a musical score. It is an example of a system that illustrates the use of a mouse to perform 2-dimensional (2D) gestures and two-phase interaction (an interaction that takes place in two consecutive phases). In this two-phased interaction, the first phase is the recognition of the intention to interact, while the second phase is the actual manipulation. In the recognition phase a gesture, such as that made when a user stops moving the mouse while still pressing the button, is recognized. In the second phase the user can manipulate parameters interactively (for example, manipulating the upward and downward note stems). An example given by Rubine (1992) is that after the *time-signature* gesture is recognised, the x and y coordinates of the mouse interactively control the numerator and denominator of the time signature.

Gscore supports the use of a variety of gestures during editing, for example gestures exist for whole, half, quarter, eighth and sixteenth notes. Although it would have been possible to have a single gesture to represent all notes and interactively control the duration and stem direction of the note, the separate gesture option was chosen to provide faster interactions. However, it is possible to edit the set of gestures and their meanings at runtime to try out various interfaces.

The creators of Gscore also developed a two-phase multi-finger interactive environment at the same time. This is a drawing program where the interaction takes place in a Sensor frame (a frame with infrared sensors to capture finger movements in space). As this system was originally developed for interaction via a mouse, the mouse operations have been mapped onto single finger gestures. Once a gesture is recognised, the other fingers are used to control additional parameters. So, for example, after a line is created, the first finger rubberbands one endpoint of the new line, and additional fingers control the line's colour and thickness.

Gscore is an example of a 2D gesture-based system that can be used to create music. It achieves this by allowing gestures to be created using an existing HCI mechanism, i.e. the mouse. As discussed in chapter 2, though this is an interaction mechanism that is possibly familiar to the users, it is still artificial and is not an everyday human communication method. The second application might allow users to perform gestures with their fingers, but the gestures produced are merely mappings of those usually performed with the mouse. For a more natural interaction approach, gestures should be performed with the hands.

Despite its shortcomings, Gscore does illustrate how gestures can be successfully incorporated into existing computing domains, such as music creation and drawing. This system is also a good example of two-phased interaction, which is the basis of an interaction dialogue.

3.2.2 Digital baton

The Digital Baton is an interface for real-time gestural control. It was originally designed by Marrin (1997) at the MIT Media Lab, as an instrument on which to perform computer music. It achieved this by replicating as closely as possible the feel of a traditional conducting baton and at the same time giving the user access to a large number of intuitive control parameters through a series of sensing systems. Many of the characteristics of hand motion are captured and used for input and control of discrete and continuous functionalities. For example, the digital baton can execute exactly-timed actions of individual notes and high level functions such as shaping volumes and coordinating separate events in time (Marrin, 1997).

The sensing systems used are an infrared LED tracking system, accelerometers, and pressure sensors. These sensors send continuous values for the baton's position, orientation, acceleration, and surface pressure to a computer via an external tracking unit. Musical conducting, being a gestural language for music, provides a good initial framework for the Digital Baton, as it is a system of mappings between specific gestural cues and their intended musical results. However the sensing technologies implemented in the Digital Baton allow it to be also mapped onto other interaction mechanisms and the system also features a 3D mouse, inertial guidance system and mini-keyboard. Figure 3.1 illustrates the Digital Baton in use.

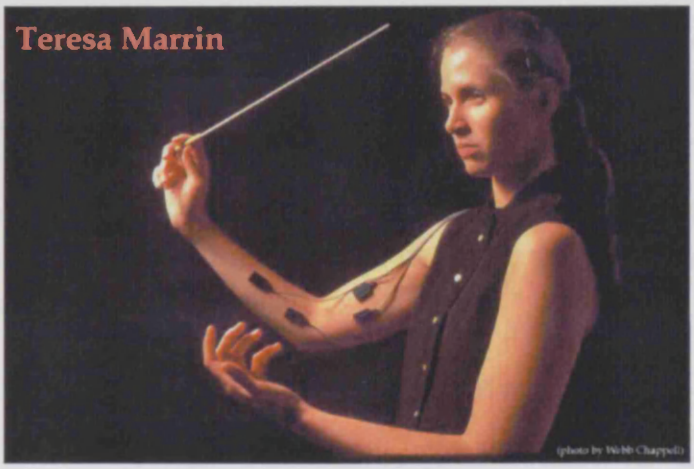


Figure 3. 1 - The Digital Baton (Marrin, 1997)

The digital baton is further example of a gesture based system and illustrates how alternative techniques can be used to capture gestures. In this case, gestures are performed using a specially designed conducting baton. It also demonstrates the mapping of gestures to intended musical results. Of particular relevance is that it is a system where the user directs system behaviour through gestures (music in this case). In principle, this is similar to the approach presented in this thesis. However, as with the Gscore system the gestures used within Digital Baton are unnatural and are mainly focused on conducting music and technically rely on the use of the baton. What is interesting with the Digital Baton is that although it was initially designed as a conducting device it has evolved into a device that can be used as a 3D mouse.

3.2.3 WorldBeat

The WorldBeat system is a two-baton gesture based system, which was designed for permanent display in the Ars Electronica Center (AEC) in Linz, Austria (Borchers 1997). The AEC is a technology "museum of the future". The purpose of WorldBeat is to demonstrate to the visitors the prospects of the use of computers in musical creativity and education, regardless of the visitor's prior computer or musical knowledge. This was achieved through a set of modules, each demonstrating a different aspect of computer use in music. Very briefly, these are:

The *Joy-Sticks* module which allows the user to play different "Virtual Instruments".

The *Virtual Baton* module through which the user may conduct a piece of music.

The *Musical Memory* module which is a game where the users have to recognize the instrument from its sound.

The *NetMusic* module where the users can cooperatively play music.

The *Musical Design Patterns* module where users can change the basic parameters of pieces of music.

The idea behind WorldBeat is to enable the visitors to control the complete exhibit using two infrared batons (see Figure 3.2). All major tasks carried out during the interaction with the exhibit are integrated into a single interface.

The visitors can use the two batons to interact with a graphical user interface, for example the virtual baton conducting module, and then use the batons to conduct the piece. This is the main difference between WorldBeat and the Digital Baton previously discussed, as the digital baton is used only for conducting purposes and not navigation.

Figure 3. 2 - WorldBeat

In order to operate the system the user stands in front of the exhibit and, watching the computer monitor, gestures with a baton in each hand (see Figure 3.2). As each baton contains infrared light emitting diodes, infrared signals are constantly emitted in all directions. Spatial signals giving the position of the baton in the form of x y b (x for horizontal position, y for vertical position and b for button press) coordinates are sent when the action button is pressed or released, while the infrared tracker, which is mounted directly below the monitor, measures the angles at which it receives the signals from the two batons. The data stream of the coordinates of the batons is sent to the base unit, where it is converted into MIDI messages. Basic gesture recognition is also incorporated in the system, which recognizes "beat" gestures as notes. This feature is used in the modules where the user plays virtual instruments using the batons as drumsticks. All events that describe MIDI playing messages are finally sent to the base unit where the requested audio signals are created and then are sent to the amplifiers, speakers, tape desk or headphones.

The right baton also functions as a pointing device, essentially acting like a 3D mouse for all navigational requirements of the system. So the user may navigate through the application by pointing at the yellow on-screen spots with the right baton and pressing the action button. However, apart from *deictic* gestures, other types of gesture are also supported by this 2-baton system. In the Joy-Sticks modules the batons are used as mallets to perform gestures commonly used to play instruments such as the drums and the xylophone. In such cases a natural mapping of the downward "beat" gestures are used to play the instruments in a velocity-sensitive way. All other interactions are again cases of "beat" gestures.

WorldBeat is a further gesture-based system that features batons. It can be used as a more general use interface for creative activities, as shown in the Joy-Sticks module. This system requires no prior knowledge of music or computing and has a very small learning overhead. This makes it attractive to the users, who are mainly children, and its usability helps attain the goal of bringing museum visitors closer to music. This system is restricted to music applications, mainly because of the use of conducting batons. The main drawback of this system is that while using a baton in the context of music might seem a natural way of interacting, it is still restrictive in that it is less suitable for use within other domains, such as History or Science, where such a mechanism is not so obviously meaningful.

3.3 Gaze and head tracking based systems

This section discusses two systems and an experiment that make use of gaze and head tracking as mechanisms for interaction.

3.3.1 EagleEyes

With the EagleEyes system, developed by Gips and Olivieri (1996), the user controls the computer through eye or head movement. The technology uses the measurement of the electro-oculographic potential obtained by using electro-oculography, EOG, through five electrodes placed on the head, to measure gaze and head positioning, as explained in Chapter 2. This system has enabled severely disabled people to control a computer. Figure 3.3 illustrates the placement of these electrodes on a user's head. The EagleEyes software acts as an interface between EagleEyes hardware and existing commercial software.

Figure 3. 3 - User with electrodes, ready to use EagleEyes (Gips and Olivieri 1996)

With the software the user can move the cursor on the screen, and perform click type selections. Like most systems that use eye or head tracking as a pointing/selection method, dwell clicking (discussed in chapter 2) is implemented. A "mouse click" event is generated when the cursor remains within a definable small radius on the screen for a certain period of time. So, by staring at a spot on a screen for a certain fraction of a second the user can operate a reasonable number of applications (an example being a web browser). "Double click" or "drag" equivalent selection method have not yet been developed, so software that requires such actions cannot be used (for example, pull down menus).

A person with no disabilities should take around 15 minutes to learn how to use EagleEyes and be able to spell their name using an alphabet board. The system measures the angle of the eye in the head, and users can move the cursor either by moving their eyes, or by moving their head while having their eyes fixed on a point on the screen, or with a combination of the two. A new user first practices moving the cursor on a blank screen and is then given a simple game to play. When the user shoots down 9 out of 10 target aliens in the game, he/she is considered sufficiently proficient to enter text and use other applications. This system has been used with dozens of people with disabilities whose learning curve, depending on their disability, may vary from 15 minutes to months.

EagleEyes enables the user to control a computer through gaze and /or head movement, by providing a software interface between existing software and the EOG hardware. The system is easy to learn and use. However, the positioning of the electrodes on the user's head is intrusive and can be very uncomfortable. In addition, as it is a system that only relies on gaze and head positioning, it is not able to support all features of existing commercial software, as such software was not designed with gaze interaction in mind. Despite this, the system does illustrate how it is possible to use alternative input mechanisms with existing software and how the gap between the two can be bridged by a software interface.

3.3.2 Head Orientation and Gaze Direction in Meetings

Stiefelhagen and Zhu (2002) carried out an experiment as part of their research to determine the object of a person's gaze, or focus of attention, from the person's head orientation. This experiment compared the results from both head orientation and eye tracking, and concluded that head orientation alone is a good indicator of the focus of attention in human computer interaction applications. Although this is an experiment that focuses on human-human interaction, the methods used, and the results are relevant to HCI. More specifically, this experiment illustrates the importance of head orientation in determining the focus of attention. This is particularly important to the work presented in this thesis as the recognition of the user's focus of attention is a key factor in the presented approach.

The experiment that Stiefelhagen and Zhu (2002) carried out was based on a scenario of a round-table discussion between four participants. The data was collected as a result of four ten minute sessions, where each participant in turn became the subject and wore a head mounted ISCAN (ISCAN) system, as shown in Figure 3.4. This system uses magnetic pose and position tracking subsystems to track the subject's head position and orientation. A head-mounted camera captures images of the subject's eyes. The software of the system can estimate and record at a 60Hz frame rate the following data about the subject:

- Head position
- Head orientation
- Eye orientation
- Eye blink
- Overall gaze/line of sight

Figure 3. 4 - Data collection with eye and head tracking system during a meeting (Stiefelhagen et.al. 2002)

From the analysis of the data, Stiefelhagen and Zhu found that in 87% of the frames, head orientation and eye gaze pointed in the same direction. In the remaining 13%, head and eye orientation were opposite. For those frames that the head and gaze orientation coincided they calculated the contribution of head orientation to the overall line of sight. A summary of the results of this experiment is presented in Table 3.1.

Subject	#Frames	Eye blinks	Same direction	Head contribution
1	36003	25.4%	83.0%	62.0%
2	35994	22.6%	80.2%	53.0%
3	38071	19.2%	91.9%	63.9%
4	35991	19.5%	92.9%	96.7%
Average		21.7%	87.0%	68.9%

Table 3. 1 - Eye blinks and contribution of head orientation to overall gaze direction (Stiefelhagen and Zhu 2002)

From the data the following interpretations were derived:

1. Most of subjects rotate their head and eyes in the same direction to look at their focus of attention
2. Head orientation varies drastically from subject to subject, although all subjects in this experiment used head orientation in at least half of their interactions.
3. Failure of the eye tracking system or eye blinks account for about 20% of the frames recorded. This is of course hardware dependent, but considering that the eye tracker used is a tracker which is used widely and considered to be one of the best wearable trackers, that 1 out of 5 times gaze direction cannot be estimated is a considerable constraint.

Stiefelhagen and Zhu (2002) concluded from this experiment that "head orientation is the most important and sometimes the only measure in gaze direction". This conclusion led them to consider how well the visual focus of attention can be predicted solely from head orientation. To address this problem, the data from the experiment was analysed to determine how often the real target person could be identified based only on head orientation. So by using the line of sight data that they had, they labelled each frame according to the person at whom each subject was looking.

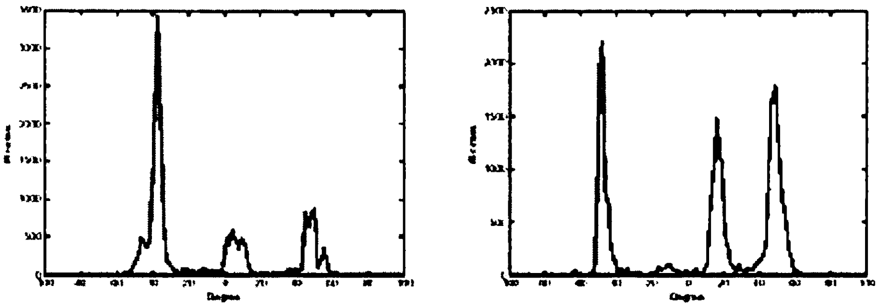


Figure 3. 5 - Histogram of horizontal gaze direction of two subjects
(Stiefelhagen and Zhu 2002)

The histograms in Figure 3.5 show the horizontal gaze direction of two subjects. Three peaks can be identified, which correspond to the directions that the other three participants are seated. From further analysis the accuracy with which focus of attention based only on head orientation, was estimated (Table 3.2).

Subject	Accuracy
1	85.7%
2	82.6%
3	93.2%
4	93.2%
Average	88.7%

Table 3. 2 - Accuracy of focus of attention estimation based on head orientation data alone
(Stiefelhagen and Zhu 2002)

From their analysis, Stiefelhagen and Zhu concluded that focus of attention can be correctly estimated with an average of 88.7%, which they view as the upper limit of accuracy. Comparing their findings to the data from the use of the eye tracking equipment they find it very impressive, as the eye tracking equipment is supposed to be more accurate.

Stiefelhagen and Zhu conclude that head orientation contributes to 68.9% to the overall gaze direction, while focus of attention estimation based solely on heard orientation can be achieved accurately by an average of 88.7% in a meeting scenario.

Stieflehagen and Zhu's experiments focused on determining whether head direction can identify the focus of attention. As identifying head orientation is much easier and in many cases, depending on the technology used, more cost effective, this experiment highlights the reliability of this approach to identifying the focus of attention. Their findings are particularly relevant as the use of head/face orientation is also examined as a possible interaction modality by this thesis (as an alternative to using gaze). The determination of focus of attention is the first phase of the interaction dialogue between the user and the approach presented by this thesis. The experimental scenario is based on a human-human interaction situation, and does not consider a HCI implementation. However, as all steps of the experiment were conducted using devices normally used in HCI, it is likely that these measurements of the human focus of attention can be applied in HCI.

3.3.3 Computer Display Control and Interaction Using Gaze: Visual Mouse

The purpose of the Visual Mouse system (Farid et al., 2002) was to develop a system for human interaction with multimedia data and multiple information streams (for example multiple video streams). This development provided a novel way to view video on demand, by only delivering video if the viewer expresses sufficient enough interest by continuing to look at the specific video stream. To achieve this the system makes use of an eye tracker connected to a PC. Figure 3.6 shows the setup of the system used by this team.

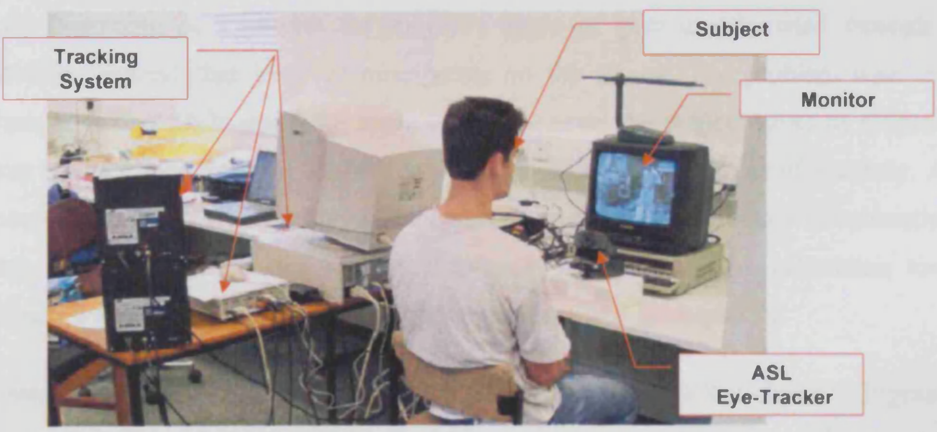


Figure 3. 6- ASL eye tracker setup for Visual Mouse (Farid et al. 2002)

This system works by reading the subject's point of gaze while he/she looks at the monitor. These are read and stored by the tracking system, so that a series of operations can be performed on this data. These operations, such as calculating the dwell time (discussed in Chapter 2) in a restricted area to emulate mouse clicks, have to be calculated in real time and then fed back to the computer system.

The Visual Mouse application demonstrates that a stream of gaze data can be used in the same way as the data that is produced by a mouse. With this application, the mouse operations are emulated and controlled by the subject's visual behaviour while observing the monitor.

Two issues that were addressed during the implementation of the Visual Mouse, and are problems that are encountered in all eye tracking applications, were:

1. The gaze coordinates are at all times subject to additional small seemingly random displacements (Mountcastle, 1980), which results in "flickering" of the coordinates. The approach that was taken to resolve this problem was to find a good compromise between the tolerance and the index reference parameters. While large values of the tolerance parameter reduce the flickering effect they also reduce the resolution of the Visual Mouse. Smaller values of the index reference parameter will generate the mouse click more rapidly, leading to the "Midas Touch" problem (Jacob, 1991) - discussed in chapter 2 - but will decrease the flickering. A possible solution to this problem is to filter the data stream of the target positions of gaze coordinates. One of the implications that arose from reducing the resolution of the Visual Mouse is that hot links should be placed on larger than normal buttons. In a test carried out with a web browser, for example, it was determined that the "back" button was too small to be invoked with the Visual mouse.

2. At the beginning of a session the subject's angle of gaze is calculated through a calibration process that involves nine points on the screen. The problem with this technique is that while using the application, whenever the subject looks at different points on the screen from the calibration points, there is some decrease of accuracy. As a solution, the designers of this system suggest an increase in the number of calibration points, but identify that this has the drawback of increasing the calibration time required, which in turn causes subject fatigue (as discussed in chapter 2).

The approach taken by Farid et al. to handling multiple streams of data is shown in Figure 3.7. A web page that shows four panels each displaying a presentation from a workshop organized by the group is displayed. There is a link to a streaming video record of each presentation. If the observer's gaze dwells long enough on a panel, the video of the respective presentation will play, until the gaze of the observer dwells on another panel.

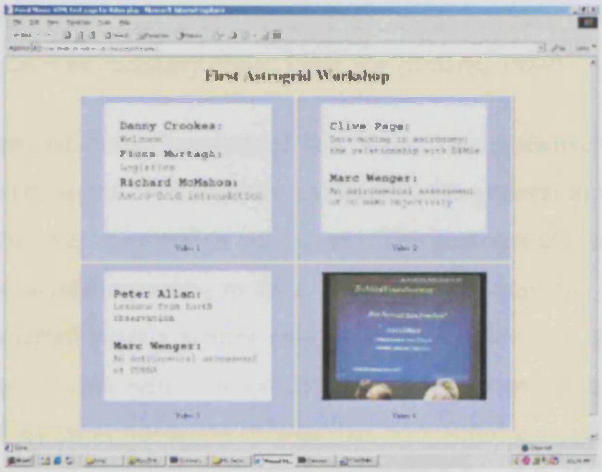


Figure 3. 7- Multiple Video streams in Browser for use with Visual Mouse (Farid et. al. 2002)

This system allows the selection of on-screen media objects through eye tracking, in a similar way to that presented in this thesis. Farid et al. also address the Midas Touch problem and suggests a possible solution. This thesis also considers Midas Touch as a problem and considers its implications in the design of eye controlled systems. Essentially, in terms of interaction, Visual Mouse is a "look and dwell" system, an eye tracking version of a "point and click" system. This type of interaction does not support the rich range of activities that users come across in their everyday interactions with commercial packages.

3.4 Multiple Modality based systems

This section discusses three systems that make use of multiple modalities as mechanisms for interaction.

3.4.1 Multimodal Natural Dialog

At MIT, Bolt and Herranz (1992), focused on incorporating human-human communication techniques into HCI. So as people communicate primarily through a combination of speech, gesture and gaze, ideally they would turn to the computer, act in essentially the same manner, and be understood. In effect, they would deal with the computer as they would another person.

Bolt and Herranz developed a system that combines speech, eye gaze input and free-hand gesture (gestures that do not require any prior vocabulary) in which both hands are used. This approach is entirely different to the one used in Gscore (discussed in section 3.2.1) which uses manual input via pointing devices (mouse, trackball, touch screen, stylus, tablet, and other similar devices). The context in which free hand is used gives the user the feeling of describing to the system in words and gestures what is to be done, instead of grasping items and trying to manipulate them. The emphasis upon two hands contrasts with studies involving one hand only, giving an intra-modality (Bolt, 1980; Hauptmann, 1989).

The gestures are interpreted, rather than treated in a "direct" manipulative way. One can think of there being a machine "agent" which somehow decodes the gestural input, and interprets it depending on what the user says and is gazing at. The gestures are not matched against templates, but rather analysed according to their "features". For example, in a case where the user wants to turn a selected block a gesture may show the rotation axis, the direction of turn and how far the object should turn. The process of interpretation starts with a parsing of speech input followed by an examination of what has been input gesturally, to complete the meaning of what has been said. Again, all objects manipulations are initiated by speech in the spirit of "co-verbal" gesturing. The hand gestures are continuously tracked by the pair of data gloves, and actions occur only when speech accompanies them.

In Figure 3.8, an example is shown in which the user performs a dual hand rotation. The objects to be manipulated are blocks and prisms of various sizes, shapes and colours. The user tells the system "Turn the block..." while with his hands the rotation axis is indicated. When there are two blocks, as in the figure, the block to be manipulated is selected either by specifying it in words or by pointing at it, or as shown in the Figure, by looking at it. This last way is the most natural of the three.

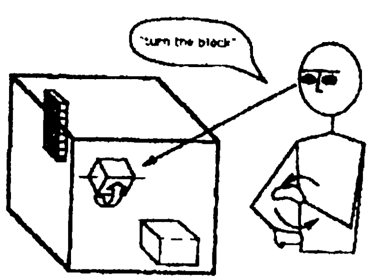


Figure 3. 8 - Rotating a block (from Bolt and Herranz, 1992)

This system is an example of a multimodal system that makes use of two-handed gestures, voice input and eye gaze in order to interact with the computer. The interaction in this system is done through the use of a "dialog" (the term "dialog" in a HCI context is discussed in Chapter 2). Such an approach is also presented in this thesis and the Bolt and Herranz system provides lessons in how to implement such an approach. The main drawback of their system is that, while it provides a gesture/voice/gaze interface, it only focuses on on-screen object direct manipulation and does not support richer interactions. Such richer interactions are more similar to those that a user performs on an everyday basis (for example, invoking and using an art package).

3.4.2 Gandalf: An Embodied Humanoid Capable of Real-Time Multimodal Dialogue with People

Gandalf is a multimodal humanoid agent, which enables full-duplex (concurrent two-way) multimodal face-to-face interaction between a human and the computer. This work was developed by the Gesture and Narrative Language Group of the Media Lab at MIT (Thorisson, 1998).

The purpose behind the construction of Gandalf was to carry out embodied, topic-oriented dialogue (for example, conducting a conversation based on the graphical model of our solar system). Gandalf is capable of conducting a fluid turn-taking and unscripted, topic-oriented dialogue with a user. To achieve this, the user interacts with the system in a multimodal way, through gestures, voice, eye tracking and posture, and the humanoid agent interacts using the same modalities. The interaction is real time and not script-based.

Figure 3. 9 - A user gets ready to interact with Gandalf

Gandalf was created using Ymir, a computational framework for psychosocial dialogue skills (Thorisson, 1996). Ymir is a hybrid, modular architecture for creating situated communicative agents. A character in Ymir (for example Gandalf) is defined by three types of processing modules: perceptual, decision and behaviour. The perception module processes the data from the sensors. The output from the perception module is then used by the decision module to make choices. These choices are then realised by the behavioural module (for example, Gandalf raises an eyebrow when greeting a person). Gandalf contains 26 perceptual, 35 decision and 83 behaviour modules (Thorisson, 1997). This structure supports the following system characteristics:

- "A character's behaviour follows common rules of turn-taking, without being rigid or step-locked.

- Gesture and facial expression are an integrated part of the communication, with no artificial communication protocols.
- Concurrent behaviours, such as glancing over to an object the speaker points at, happen naturally and where they are expected.
- When speech overlaps or miscommunication occurs, it is dealt with in the same ways as in human face-to-face interaction, by stopping, restarting, hesitating, etc." (Thorisson, 1997 p536)

This system is an example of a system that uses alternative interaction modalities for the user to interact with the computer and the computer to interact with the user, implementing a multimodal dialog. The purpose of this system is to support the user in interacting with the agent. It does not allow the user to perform any other activities using these modalities, nor does this interaction through the agent invoke any other task or activity. However, it does provide an example of how two-way HCI using alternative interaction modalities can be achieved.

3.4.3 Manual And Gaze Input Cascaded (MAGIC)

Manual and Gaze Input Cascaded (MAGIC), discussed by Zhai et al. (1999), is a system where the pointing and selection of on screen objects is done manually, but is supported by the use of gaze tracking. The idea behind the system is to use gaze to dynamically redefine (or warp) the 'home position' of the cursor pointer to the screen area close to a target at which the user is looking. The intention is to reduce the amount of manual movement required of the pointer. So, for example, when a user operating MS Windows looks at the "My Computer" icon the mouse pointer will move close to the icon so that if the user want to select it, then this can be done with only a small amount of mouse movement. According to Zhai et al. (1999) the identification of the user's intended target is critical to the effectiveness of such a system.

The team designed two MAGIC systems; one liberal and one conservative in terms of the relationship of target identification and cursor placement. When the liberal approach is used, the cursor is warped to every new object that the user looks at (Figure 3.10). The user may then use a manual pointing device, such as a mouse, to take control of the cursor which is already either near or on the target. Alternatively, they may ignore it and look at another target. A new target/object is operationally defined as being at a sufficient distance from the current cursor location. This distance threshold is set to 120 pixels, which prohibits the cursor to warp when the user does continuous manipulation using the pointing device, in cases such as drawing. This approach is characterized as "pro-active", as the cursor is always waiting in the vicinity of any potential target.

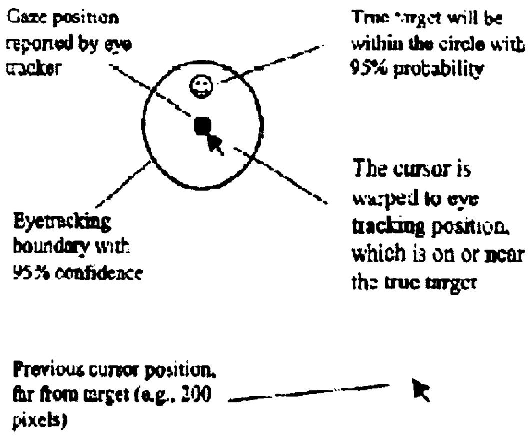


Figure 3. 10- MAGIC: the liberal approach taken from Zhai et al. (1999)

The conservative approach is different in that the cursor does not warp to each target/object that the user's gaze focuses on, but, instead, waits until the user moves the pointing device (Figure 3.11). This movement of the pointing device triggers the system to warp the cursor pointer to the gaze area that the eye tracker has identified. Again, the user then needs to make a small manual movement to bring the cursor on the exact target.

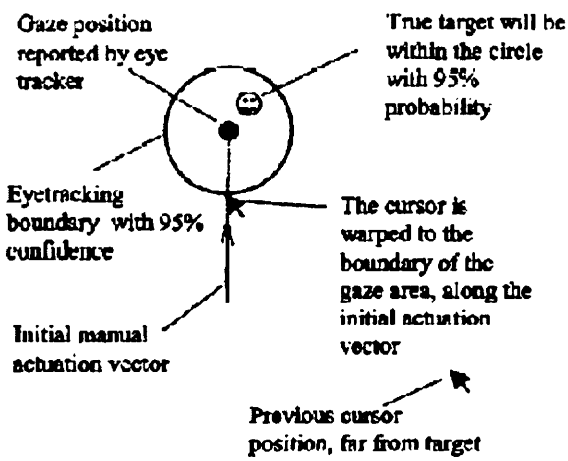


Figure 3. 11 - MAGIC : the conservative approach taken from Zhai et al. (1999)

MAGIC is a system that combines gaze input and 2D hand gestures performed with a mouse to assist users in selecting objects on the screen. Of particular relevance is that this system also features a form of two-phased interaction, where the users enters into a dialog situation by dwelling his/her eye gaze on a screen object and then the system responds by warping the pointer to the vicinity of the object. The work in this thesis also proposes a two-phased interaction technique and MAGIC helped to inform its design in this respect.

The main advantages of MAGIC are its ability to reduce the distance required for mouse movement due to the use of pointer warping. Consequently, it is possible that MAGIC pointing can be considerably faster than manual pointing. In addition, MAGIC also possesses a software interface that allows it to be used with existing applications that run on MS Windows NT.

The main drawback of this system is that it uses mouse gestures, and not hand gestures, limiting the possibilities of gestures and confining them to 2D. As MAGIC is not used for systems that have been designed for gaze / gesture input, it may be less intuitive to use existing applications in this multimodal way.

3.5 Discussion

This chapter has examined key systems that have been developed that also use gaze, gestures, or a combination of both, as a means of interaction (a summary is provided in Table 3.3). Analysis of these systems was influential in the design of the Conductor Interaction Method.

The Conductor Interaction Method that is presented in this thesis features two interaction modalities; gaze and (hand) gestures. More specifically, the method uses two-phased interaction in which gaze is used for *selection* activities while gestures are used for *manipulation* activities.

Both the Multimodal Natural Dialog and the Gandalf systems make use of gaze and hand gestures (with the former also using a similar two phased interaction method as embodied in the Presentation Conductor) and so the experience and techniques developed with these systems helped inform the design of the Conductor Interaction Method. The limitation of both these systems, however, is that ultimately the interaction is simplistic; with the users being unable to perform the complex manipulations intended for the Conductor Interaction Method. The MAGIC system also uses a two-phased interaction method, but it uses mouse, rather than hand, gestures to perform the manipulation.

The use of gaze as a method for selecting entities has also been explored within the EagleEyes and Virtual Mouse projects. Of particular interest is the investigation of the Midas Touch problem, by these two projects, and the suggested ways to tackle it. This is an issue that the Conductor Interaction Method addressed, and their experiences were drawn upon for tackling this.

Two of the examined systems (MAGIC and EagleEyes) support alternative interaction modalities (gaze and gestures) being used to control existing applications that are not geared towards such modalities (for example, a word processor). Although they provide satisfactory results, ultimately they illustrate the need for interfaces that are tailored towards the specific modality that is to be used. This supports the argument, made by this thesis, for the need to develop new interaction methods if gestures and gaze are to be used to interact with a computer.

Because, the Conductor Interaction Method (and in particular, the Presentation Conductor system) is most likely to be used for creative activities, a key issue is whether the use of gaze and gestures can help to promote creativity within a system whilst at the same time not requiring a large learning overhead in order for the user to master them. The Gscore, Digital Baton and WorldBeat systems have all illustrated that gestures can support creative interaction. WorldBeat also demonstrates that using familiar gestures to perform specific tasks can significantly reduce the learning overhead required by the user. It was desirable for the Conductor Interaction Method to also have a low learning overhead and drawing upon the familiarity of certain gestures and their mappings to specific tasks in the users' everyday lives, can be one way to achieve this. The experiences of the WorldBeat system helped inform the design of the types of gestures that can be used.

Additionally, the use of the Conductor Interaction Method with large visual displays was investigated (as discussed in chapter 6). In such an environment the use of face/head direction as a means to discover a users focus of attention may be a more suitable alternative to measuring a users gaze. The experiment carried out by Stiefelhagen and Zhu (2002) illustrates how head direction can be successfully used to determine focus of attention, and so demonstrates that a face/head direction interface is a valid alternative to experiment with.

3.6 Chapter Summary

This chapter has examined nine key systems that use eye gaze, gestures or a combination of both as a means to interact with a computer system. Analysis of these systems has highlighted issues that informed the design of the Conductor Interaction Method and accompanying Presentation Conductor system that was developed. Of particular relevance is the experience and methods gained for tackling the Midas touch problem, as well as how two phased interaction can be supported. Both of these are issues that are addressed by the Conductor Interaction Method.

In the next chapter the Conductor Interaction Method is presented, along with an architecture for the Presentation Conductor system that seeks to realise and evaluate this method. A detailed description of the method including an analysis of the metaphors used and their significance for its functionality is provided. The architecture of the prototypical Presentation

Conductor system is then presented, along with a scenario based worked example of how the architecture functions.

System	Interaction techniques used				Purpose of use	Relevant properties of system
	Voice	Eye gaze	Hand Gesture	Other Gesture Device		
Gscore			*	Mouse	Music creation and painting	Use of gestures in creative applications
Digital baton			*	Digital baton	To conduct music	Use of a special purpose 3D gesture capturing device to recognise conducting gestures
WorldBeat				Pair of batons	Interaction in a creative – educational music application	A simple interface for users with no prior knowledge
EagleEyes		*			Interaction in general purpose Interfaces, mainly for disabled	The use of EOG to record eye – head movement
Head Orientation and Gaze Direction in Meetings		*			To investigate the accuracy of determining the focus of attention from head movement vs. eye gaze	That head orientation – facial pointing is in the majority of cases sufficient to determine the focus of attention
Computer Display Control and Interaction Using Eye-Gaze		*			To use eye gaze to select on screen media objects	a. Eye tracking equipment may need to be recalibrated depending on the original calibration b. Dwelling as a selection method has many problems (Midas Touch)
Multimodal Natural Dialog	*	*	*		Interaction with on screen objects	An implementation approach
Gandalf	*	*	*	Body suit	To conduct embodied multimodal dialogue	How two-way multimodal dialogue can be achieved
MAGIC		*	*	Mouse	To facilitate interaction of traditional mouse based interfaces	The combination of eye tracking and 2D gesture (mouse based) interaction

Table 3. 3 - A summary of systems reviewed in this chapter

Chapter 4 – Designing the Conductor Interaction Method and Presentation Conductor

4.1 Introduction

This thesis presents an argument for the use of an alternative, more natural, Human-Computer interaction method. In chapter 2 the failings of existing HCI techniques were highlighted. It was argued that interaction techniques that humans are more familiar with, such as gaze and gestures, can also be beneficial and should be considered as a means of interacting with a computer.

Chapter 3 examined key systems that have been developed that use eye gaze, gestures or a combination of both, as a means to interact with a computer. Analysis of these systems identified a number of issues and experiences than have helped inform the design of the proposed Conductor Interaction Method. In particular, in this work we seek to address the Midas Touch problem, as well develop a system that utilises two-phase interaction.

This chapter presents the Conductor Interaction Method that is proposed by this thesis. It begins by first discussing the use of Metaphors within HCI. The nature and functionality of metaphors is examined and an overview is provided of the process of metaphor creation within HCI. The chapter then moves on to discuss two novel metaphors, the Orchestra and the Conductor metaphors, that have been developed and which form the basis of the proposed Conductor Interaction Method. A description of the Conductor Interaction Method and its key features is then provided.

In order to illustrate the Conductor Interaction Method in use and also to provide a mechanism for its evaluation, a novel system, the Presentation Conductor, has been developed that utilises the Conductor Interaction Method. This chapter presents the architecture for the developed Presentation Conductor, as well as providing scenarios that illustrate how the method and system will work in practice.

An implementation based on this architecture is presented in chapter 5 and this, and the interaction method, are evaluated in chapter 6.

4.2 Metaphors in HCI

A general definition of a Metaphor is "*a figure of speech in which a word or phrase literally denoting one kind of object or idea is used in place of another to suggest a likeness or* *analogy between them*" (Merriam-Webster Dictionary). As this definition is specifically related to language it cannot be directly applied to User Interfaces without re-defining it (Pirhonen, 2001).

Lakoff and Johnson's work provides a basis for understanding the use of non-verbal metaphors in HCI. In their theory (Lakoff et al., 1980), the potential of a metaphor is shown in its power to enhance the understanding of new concepts in terms of familiar ones. For example, the image of a folder is used to represent a directory in a file system. According to this theory, metaphors are conceptual in nature, and metaphorical language (for example, the expression "The Iron Lady" being used to refer to Margaret Thatcher during her years as Prime Minister of Britain) is only one form of expression for these conceptual entities. From this it is easy to understand the use of non-verbal metaphors.

Metaphors are not a novel feature of HCI in themselves, the *desktop metaphor* (Xerox) being a prime example, in which users seemingly use a virtual desktop to organise and interact with their computer. Metaphors are used in computer interfaces as a means of assisting the users in understanding a new target domain by associating it to a source domain that they can usually understand or are familiar with (Baecker et al., 1995). For example, in a WIMP interface running an application (target domain), is associated with opening a window (source domain).

Since the appearance of Macintosh-style GUI, the term metaphor has been used widely in the context of user-interfaces. Even though words such as the term "file" were already being used, it was the visual expression that strengthened the analogy between physical file and computer file (Pirhonen, 2001). This affects the way users understand metaphors by emphasising similarities and leaving the analogy at an abstract level. So, while traditional metaphors are based on analogies, visual metaphors in GUIs typically rely on visible similarities.

In HCI, metaphors can be used to help users understand concepts and interaction styles, but it has also been argued that they can have a negative impact on interfaces, because they may impose too many constraints on the design space that interface developers can use (Cooper, 1995).

In order to gain a better understanding of the functionality and use of metaphors, it is necessary to understand the process of metaphor creation.

4.2.1 Metaphor creation

In Umberto Eco's discussion on Aristotle (Eco, 1984 p101), he points out that Aristotle describes the creation of metaphors as " 'a sign of natural disposition of the mind' because knowing how to find good metaphors means perceiving or grasping the similarity of things between each other". Adopting this view in HCI, i.e., finding common elements such as functionality or interaction style, can prove to be a good starting point in constructing a metaphor.

Caroll, Mack and Kellog (1988) have identified three approaches for metaphor design:

1. *Operational* approaches, which focus on how and to what extent metaphors have a measurable effect on learning. For example, when referring to text editing an analogy can be made between a conventional folder and a computer directory, pages can be added and removed to the folder, just as files can be added or deleted from a computer directory.
2. *Structural* approaches, represented by Gentner's (1983) structure mapping theory, where metaphors are examined by developing formal representations (such as a graph) in the source domain and the target domain. A typical example that Gentner uses to illustrate this is Rutherfod's analogy between the solar system and the structure of the hydrogen (H) atom. The sun and planets of the solar system (source domain) map to the atomic nucleus and the electrons of the H atom (target domain). As the sun is the centre of the solar system and is larger than any other planet, the same holds for the nucleus. But, as Gentner points out, all characteristics do not map successfully. For example, in this case, the sun is yellow and hot, but the H nucleus will not necessarily be.
3. *Pragmatic* approaches, which acknowledge that in real-world situations, metaphors inevitably involve incompleteness and mismatches, and that the power of metaphors may actually lie in such inconsistencies between the source and the target domains. For example, the use of a TV broadcast metaphor to represent a one way, one-to-many communication.

Each of these three approaches focuses on a very specific area of metaphor design, that have their own strengths and weaknesses. For this reason, it is logical to adopt a synergetic strategy that attempts to bring these three approaches together when designing metaphors (Ramloll, 2000).

According to Madsen (1994) there are three main activities carried out during metaphorical design: the *generation*, *evaluation* and *development* of metaphors.

- **Generation of Metaphors**. Issues that have to be considered during this activity include:

  Extension of established metaphors. By building on already established metaphors, the chances that users will assimilate the metaphor are high. Also, metaphors that reflect a physical structure are often successful (Caroll, 1982).

  Audience Background. It is important to consider the background knowledge of the user, as this comprises the source domain.

- **Evaluation of Metaphors**. A number of criteria can be used to evaluate metaphors, some of which are:

  The Structure of a metaphor. One issue related to the structure is how semantically rich the metaphor is. For example a "TV-Broadcast" metaphor used for data transmission, provides a rich semantic background such as stations, channels receivers etc. (Madsen, 1994). Another issue related to the structure is its applicability. For instance, in the "TV-Broadcast" example the metaphor implies that the data is transmitted instantaneously, which might not be the case.

  Examining the formal mapping between the source and target domains (Ramloll, 2000). The source domain is what the user already understands and the target domain is the more complex domain that with the use of the metaphor should be explained in terms of components of the source domain. With a formal examination of the mappings of these two domains one should be able to identify whether a reasonable number of analogies can be drawn between the source and the target domains. The fewer number of analogies, the less appropriate the metaphor, while if the analogies are significant the metaphor is well grounded.

  Effect of learning (Ramloll, 2000). The way that a metaphor affects the learning curve in the use of an interface is significant in evaluating a metaphor. The smaller the learning curve, the better the metaphor. The speed with which the user completes set tasks is also related to the appropriateness of the metaphor.

- **Development of Metaphors.**

  This stage mainly involves studies of how the metaphors function in the application, and refining them. This is necessary so that designers may assess whether there are more design iterations needed and whether there are problems that have not been identified earlier, and may lead to rejecting the metaphor (Ramloll, 2000).

The design and development of metaphors for HCI is a complex task and this section has provided an overview of the key aspects involved. In order to support the proposed Conductor Interaction Method, two metaphors have been developed, the Orchestra and the Conductor metaphors. These metaphors are discussed in the following sections.

4.2.2 Orchestra Metaphor

In order to provide the user with an environment in which they can interact using gestures and gaze, an orchestra metaphor has been developed. The purpose of this metaphor is to graphically present to the user the resources that are available for them to manipulate. The graphical representation of the resources and their positioning is done in a functional way to allow the user to visually recognise them, and to interact with them. For example, the image of a jukebox is associated with music, therefore inviting the user to interact with it to play some music. Because these resources can be visually recognised by the user, this metaphor can be used in gaze based interaction.

The orchestral metaphor has its origins in the set-up of a theatre; the stage is the main area where the play is performed, and below this is the orchestra that consists of musicians playing their musical instruments under the guidance of a conductor, to supplement the play. The orchestra metaphor follows a similar idea to the theatrical set-up. It uses the stage as the main area of interaction and presentation, and the orchestra represents the resources that are available to the user. There are no actors or musical instruments, but "media objects", which are visual representations of the media galleries (the groups of available resources) that they represent. These objects are placed on the two sides of the stage (figure 4.1), leaving the centre stage available for the displaying and manipulation of these objects (for example, combining them into a presentation). The functionality of the centre stage is based on the actual metaphor, as the user should expect whatever happens to take place in that area of the screen. For example, if the user chooses an animation clip, it will be displayed in the central stage area.

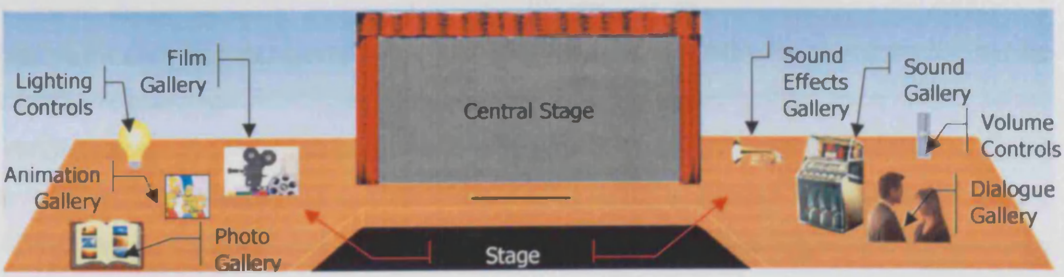


Figure 4. 1 - The Orchestral Metaphor

Figure 4.1 provides an example of the orchestra metaphor with media object galleries that could be used for the creation of multimedia presentations, and media controls that can be used to manipulate the media objects. For example, the Photo Gallery, depicted by a picture book, could represent a selection of images stored in image format types such as jpeg, gif, bmp etc., which the user can use in a presentation. The Lighting Controls, depicted by a light bulb, could allow the user to adjust the brightness or contrast of the image.

The orchestra metaphor can provide a number of useful features. In particular, visual representations of the resources are used, so gaze based interaction can be exploited. Additionally, the familiarity of the stage set-up gives the user the sense of expectation and helps in orienting him/her with the interface.

In order to interact with the orchestra metaphor it is proposed to use a *Conductor metaphor* as is now defined.

4.2.3 Conductor Metaphor

The conductor metaphor is used in conjunction with the orchestra metaphor, and is predominantly an interaction metaphor. Just as a conductor can interact with the musicians of an orchestra, the conductor metaphor allows a user to interact with the media objects and controls that are represented within the orchestra metaphor.

As the metaphor's name suggests, the user interacts with the media objects in the same way as a conductor would interact with the musicians of an orchestra, i.e. establishing eye contact to initiate the interaction, and then using bimanual gestures to specify when and how the musicians will play. As a music conductor is silent throughout the interaction, but uses body language to convey information to the members of his orchestra, so the user is able to use silent interaction to manipulate the media objects and controls. By breaking down the interaction of the music conductor, it is easy to identify the analogies between the way the music conductor and the user in the case of the orchestral metaphor interact. The music conductor has the orchestra in front of him/her; the user has the media galleries and controls in front of him/her. The music conductor establishes eye contact with the musicians that he/she prompts to play their instrument; the user looks at the media gallery from which he/she intends to select an item. The music conductor uses bimanual gestures to guide the musician through the performance; the user uses bimanual gestures to interact with the media gallery objects. For example, when presented with a setup similar to that in Figure 4.1, the user may look at the "picture book" to select the photo gallery. Then using hand gestures, the user may select a picture of a forest fire and with the appropriate gesture position the picture on the central stage area where it will eventually be displayed.

The Conductor metaphor itself provides a number of advantages. In particular, it is an interaction metaphor that has been designed specifically for gesture and gaze based interaction methods, and so is more suitable than other existing metaphors such as the desktop metaphor. The metaphor is also designed to be used in tandem with the orchestra metaphor. It is believed that, together, they can provide an interface that is easier to use and understand, especially for novice computer users.

4.3 The Conductor Interaction Method

This thesis proposes a novel interaction method, the Conductor Interaction Method, which aims to provide users with a more natural way of interacting with a computer. This method features both gaze and gesture interaction, and also uses the Orchestra and Conductor metaphors in order to help the user understand and exploit the interface. It is envisioned that this method will be most beneficial for users who have little experience with computers, enabling them to manipulate media objects easily. It could be used in a number of areas, including creative domains (for example, where the user may create presentations, music, stories, lessons, etc.), and control domains where the user may control an environment (for example, the lights and sound in a house or a night club, or a surveillance system).

More specifically the Conductor Interaction Method aims to provide:

- *A more natural interface that utilises gaze and gestures, but is nevertheless capable of supporting sophisticated activities.* The Conductor method aims to provide an interaction technique that is as natural as possible and is close to human-human interaction methods with which users are already familiar. In order to achieve this, the method makes use of gaze and gesture modalities as interacting mechanisms. The combination of both allows the user to perform not only simple interactions with a computer but also more complex interactions such as the selecting, editing, and placing of media objects.

- *An interface that uses the Orchestra and Conductor metaphors.* The Conductor method uses these metaphors to encourage the user to interact with the system in a multimodal way. Through the simple Orchestra interface the user/conductor can modify the available resources in ways that the user can relate to. This is particularly important for those users who have little or no experience in computer use.

- *A two-phased interaction method.* The Conductor method uses an interaction process where each modality is specific and has a particular function. The interaction between the user and the interface can be seen as a *dialogue* that is comprised of two phases. In the first phase, the user selects the on-screen object by gazing at it. In the second phase, with the gesture interface the user is able to manipulate the selected object. These distinct functions of gaze and gesture aim to increase system usability, as they are based on human-human interaction techniques, and they also help to overcome issues such as the Midas Touch problem (discussed in chapter 2). As the dialogue combines two modalities in sequence, the gaze interface can be disabled after the first phase. This minimises the possibility of accidentally selecting objects through the gaze interface. The Midas Touch problem can also be further addressed by ensuring that there is ample 'dead space' between media objects.

- *Significantly reduced learning overhead compared to existing interaction methods.* The Conductor method aims to reduce the overhead of learning to use the system by encouraging the use of gestures that the user can easily associate with activities they perform in their everyday life. This transfer of experience can lead to a smaller learning overhead (Borchers, 1997), allowing users to make the most of the system's features in a shorter time.

In order to help demonstrate the Conductor method this thesis also presents the development of a system that uses the method as a means for human-computer interaction. Not only does this help to realise and demonstrate the method, but it also facilitates an evaluation of the method.

The rest of this chapter presents the architecture for the Presentation Conductor system. This system enables the user to create presentations by utilising the Conductor method. After the architecture has been described a scenario is provided that illustrates how the Conductor method and the Presentation Conductor system would function.

The implementation of the Presentation Conductor is presented in chapter 5. An evaluation of the method and developed system is presented in chapter 6.

4.4 An Architecture for The Presentation Conductor

The Presentation Conductor is a system that allows users to construct and display multi-media presentations via the use of the Conductor interaction method that has been presented in the previous section.

Existing tools for creating multi-media presentations typically possess complex interfaces and require a large learning overhead for the inexperienced user. The proposed Presentation Conductor system aims to employ the Conductor method and in doing so seeks to provide an interface that is more akin to every day human-human interaction methods. It is believed that such an interface will be more natural, understandable and easier to use for less experience users.

The Presentation Conductor implements the Conductor method and the Orchestra and Conductor metaphors that comprise it. Through gaze and gesture interfaces the user (conductor) creates presentations by selecting and manipulating Media Objects. As discussed in 4.2.2, these Media Objects are grouped within Media Galleries and include images, videos, animations and sounds. Depending on its type, each Media Object has a specific set of properties that can be modified by the user using the appropriate Media Controls.

The 'central stage' aspect of the interface represents the users work area and is where the selected Media Objects are manipulated, and the presentation is displayed. Constructed presentations may be previewed at any point, or stored and displayed at a later date.

This section describes the architecture of the Presentation Conductor that that has been designed. An overview of the architecture is initially presented followed by a more detailed description of its components. Section 4.5 provides an example that illustrates how the architecture and the Conductor method can be realised in a working system.

4.4.1 Overview

The architecture of the Presentation Conductor system consists of three major components and is illustrated in figure 4.2:

- The ***User***, who interacts with the system to create a presentation and represents the conductor, in the conductor metaphor (c.f. 4.2.3).
- The ***Interfaces***, that enable the user to interact with the application. There are three interfaces, the gesture interface and the gaze interface, through which the user provides input through the respective modality. These interfaces do not provide

feedback from the application to the user. The feedback from the application is given via the third interface, the audio-visual interface.

- The ***Application***, enabling the user to create a multimedia presentation by interacting through the gesture and gaze interfaces. The application utilises the orchestra metaphor to support the user in creating a presentation. A visualisation of the metaphor is presented to the user via the computer display. (c.f. 4.2.2). The Application also manages the stored media objects and previously saved presentations.

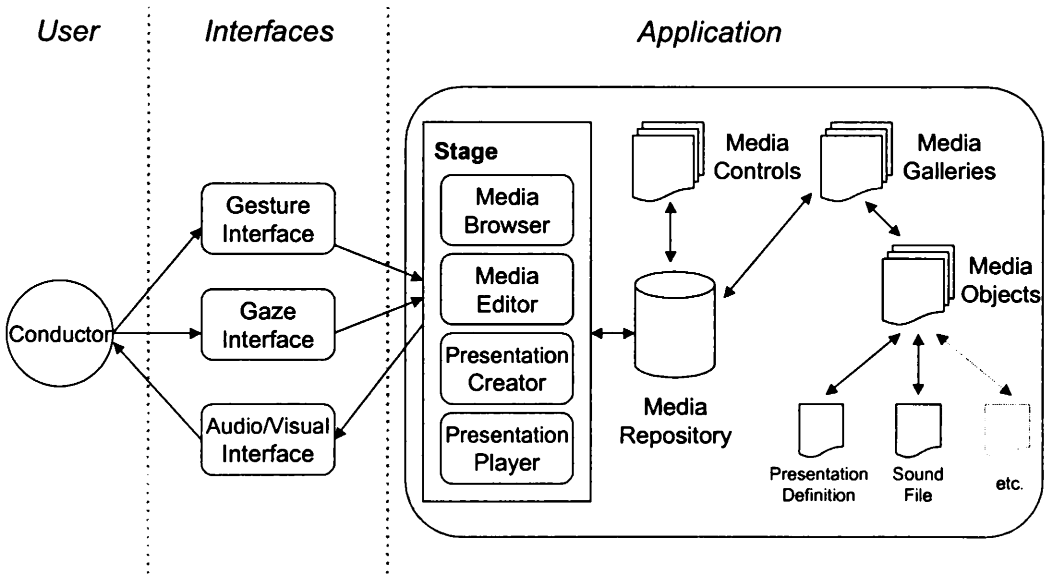


Figure 4. 2 - The Presentation Conductor Architecture

The following sections discuss the components of the architecture in more detail.

4.4.2 The User

The user, as previously mentioned, represents the *Conductor* from the conductor metaphor. He/she interacts with the system to create a multimedia presentation through the gesture and gaze interfaces, while feedback is given on a display and through speakers. The Conductor decides on the structure of the presentation and which media objects from those stored will be used to construct the presentation. The conductor may also modify certain properties of the media clips. Finally, it is the Conductor who decides to store or discard a presentation.

4.4.3 The Interfaces

The architecture has three types of interfaces to support user interaction with the application:

- The Gaze Interface

- The Gesture Interface
- The Audio/Visual Interface

4.4.3.1 The Gaze Interface

The gaze interface allows the conductor to interact with the application through gaze. Gaze itself is not used to issue complex commands, but rather to select elements from the visual display. The gaze interface recognises the user's focus of attention on a specific object, and then informs the application to activate it. When used in conjunction with the gesture interface the conductor is able to select and manipulate an object.

4.4.3.2 The Gesture Interface

The gaze interface allows the conductor to interact with the application through gestures. Within the Presentation Conductor, gestures are used for complex manipulations, in particular manipulating the media objects and media controls.

Gestural interaction can only occur after an object has been activated through the gaze interface. The object can then be manipulated through a series of gestures, which are carried out by both of the conductor's hands. The gesture interface recognises the gestures that are being performed, and communicates this to the application. All the interaction between the conductor and the application from the moment that a specific object has been activated (via gaze) until the moment that the conductor has finished manipulating the media object, is carried out gesturally. The gesture interface is able to recognise a small vocabulary consisting of emblematic and pantomime gestures (these gesture types were described in chapter 2).

4.4.3.3 The Audio/Visual Interface

The audio/visual interface provides feedback to the conductor from the Presentation Conductor. The main function of this interface is to display the visualisation of the orchestral metaphor (for example, on a computer monitor or projection screen) and to provide audio output. Any feedback from the application is relayed to the user through this interface. This includes the preview of the media objects and the visualisation of the presentation, both for the preview and final playing of the presentation. Feedback is not provided by the gaze and gesture interfaces.

4.4.4 The Application

The conductor can create, manipulate and present multimedia presentations with the application. As shown in figure 4.2, the application focuses on providing a 'stage' that acts as the conductor's work area, as well as managing a media repository that stores the media galleries, their media objects, and previously created presentation definitions.

The Application itself is comprised of five modules that handle the different aspects of the multimedia presentation creation process, as well three data objects that capture information that is used by the system:

The five modules are:

- Media Browser
- Media Editor
- Presentation Creator
- Presentation Player
- Media Repository

The data Objects are:

- Media Object
- Media Gallery
- Media Controls
- Presentation Definitions

The data objects and then the modules are described in the following sections.

4.4.4.1 Media Object

A Media object represents a stored file that contains multimedia data. This could be for example an image (JPEG, GIF, and so on), sound (WAV, MP3, etc.), video (MPEG, AVI, etc) or even stored presentations (represented by Presentation Definitions, discussed later). However, alternative media types could readily be incorporated should they become available.

4.4.4.2 Media Gallery

A Media Gallery is essentially a directory of media objects of a specific type (for example, images) or with specific characteristics (for example, distinctive sounds such as a siren or breaking glass). The purpose of the Media Gallery is to categorise media objects so that the Conductor can easily navigate through them. Individual media galleries are visually represented on the stage. Table 4.1 list the properties of a Media Gallery object.

Properties	Description
Name	A name that is indicative of the contents of the Gallery (for example, *Photo Gallery*)
Visual Representation	The visual representation for the Media Gallery that is displayed to the conductor on the Stage (for example, a picture book)
Collection of Media Objects	The stored files that contain multimedia data of the specific type suitable for the Media Gallery (for example, a JPEG image)

Table 4. 1 - Media Gallery Properties

4.4.4.3 Media Controls

The Media Controls are objects that represent a control of specific properties of the various media. Each type of media object has different properties that can be modified through a Media Control. For example, a picture may have its brightness and contrast modified while a sound may have its volume and pitch. A mapping exists between Media Objects and Media Controls, so that the correct Media Control is applied to a Media Object. An example of a mapping technique that could be used is the use of file types as an indicator of the Media Object type. For example, a volume control can be associated with sound media objects with a file extension of .wav. Table 4.2 illustrates the properties of Media Controls.

Properties	Description
Name	A name that is indicative of the property that the Control modifies (for example, *Brightness*)
Visual Representation	The visual representation for the Media Control that is displayed when it appears to the conductor on the Stage (for example, a light bulb)
Value	The values that the property can posses. This could be, for example, a range of values (such as 0-100 for brightness) or Boolean values (such as True or False for looping of a sound)
Mapping	The mapping between the Media Control and the Media Objects that exists (for example Brightness applies to pictures, animations and videos). This could be based on file extensions.

Table 4. 2 - Media Control Properties

Media Controls are individually visually represented, and appear when the relevant Media object is selected. They are not organised into galleries.

4.4.4.4 Presentation Definitions

The Presentation Definitions are objects that capture the structure of a presentation (for example, the media objects used, their location, etc). The application is able to save these definitions as a file, thus allowing the presentations to be stored and re-edited at a later date (similar in manner to a PowerPoint presentation file). However, as well as being a mechanism to capture and store created presentations, it also allows for created presentations to be incorporated into new presentations. For example, a new presentation could be created that incorporates one or more previously created presentations.

A Presentation Definition is comprised of properties that represent the Media Objects in the presentation. Each packet stores information about the Media Object and its role in the presentation as shown in Table 4.3.

Properties	Description
Media Object Identifier	A unique Identifier for the Media Object in the presentation
Media object type	The type of the Media Object. For example, sound effect, image, etc
Location of Media Object	The location of the Media Object. This could, for example, be a filename or a URL.
Position of Media Object	The position of a *visual* Media object in a presentation, i.e. its location on the visual display
Timing of Media Object	When the Media Object is scheduled to appear in the presentation and its duration
Modifications	Properties of the Media Object that have been changed by Media Controls

Table 4. 3 – Presentation Definition Properties

4.4.4.5 Media Browser

The Media Browser module enables the conductor to browse through the Media Galleries that are available to the application. By using both gaze and gestures the conductor is able to select and manipulate Media Galleries and preview the Media Objects they contain. Gaze can be used to select the desired Media Gallery visualisation that exists on the Stage and then the conductor can use gestures to choose the Media Object from that gallery. Once a Media Object has been selected it is passed on to the Media Editor where it can be tailored for the presentation.

4.4.4.6 Media Editor

The Media Editor module enables the conductor to apply the Media Controls to the selected media object. Depending on the type of Media Object different controls may be applied. For example, a volume control could be applied to a music file. Depending on the number of relevant Media Controls, a combination of gaze and gestures would be used to first select the desired media control and then to edit the media object's properties.

4.4.4.7 Presentation Creator

The Presentation Creator module deals with the actual generation of the multimedia presentation, as well as handling its storage. It is this module where the actual presentation construction takes place and it is therefore the main module within the application. From here the conductor can browse the Media Galleries, edit Media Objects, store/view presentations, etc.

The Presentation Creator module itself has three main functions:

- *Positioning of Media Objects on the central stage area.* When the user has completed the editing of the Media Object's properties, the Presentation Creator positions the Media Object on the central stage (where other, previously positioned, objects may exist). With gestures the user may position the Media Object anywhere on the central stage for the purpose of the presentation.

- *Handling of Media Object timing.* The Presentation Creator also handles the sequence and the lifetime of the Media Objects that exist in a presentation. A lifetime spans from the moment that a Media Object appears in the presentation until either:
    1. The presentation ends.
    2. Until the Media Object is removed as part of the presentation sequence.
    3. Or, the Media Object has a time property that has expired (for example, with sound and video clips).

  The Presentation Creator also keeps track of a presentation's sequence, which represents the points in the presentation when the different Media Objects come into play.
- *Storing of presentations in the media repository.* The Presentation Creator is able to store created presentations in the Media Repository. This involves the creation of a Presentation Definition object, which is then stored.

4.4.4.8 Presentation Player

The Presentation Player module is used to preview or play a presentation that is currently being edited. The conductor is able to move backwards and forwards through a presentation as well as stop it at any point, by using gestures.

4.4.4.9 Media Repository

The Media Repository module stores the data that is used by the system. It stores the Media Galleries, the Media Objects that make up the galleries and the Media Controls.

4.5 Scenarios of the Presentation Conductor Architecture and Conductor Interaction Method in use

This section provides three scenarios to illustrate the operation of the Presentation Conductor architecture as well as to demonstrate the suitability of the Conductor interaction method. These scenarios involve a hypothetical user, Jack, who seeks to create a presentation that comprises of an image and sound clip. He views and saves his presentation, and then browses and loads a previously created presentation.

A detailed description of the gestures comprising the gesture vocabulary is presented in Chapter 5.

Within these scenarios the Media Repository contains Photo, Sound, Video and Presentation Media Galleries. Appropriate Media Controls also exist for the respective Media Objects.

4.5.1 Selecting, editing and placing an image and sound clip within the presentation

In this scenario Jack uses the Presentation Conductor system to select an image from the Image Media Gallery. He then adjusts its brightness and contrast, before placing it in the correct position within the presentation. He then performs the activity again, but this time adds a sound clip to the presentation. He adjusts the volume and pitch of the sound clip.

Steps:

1. Jack starts the Presentation Conductor system. The Presentation Creator module is activated and the stage is displayed with the "orchestra" of Media Galleries placed on the two sides of the Central Stage. He fixates his gaze on the "picture book" icon. The Gaze Interface interprets this action, and informs the Presentation Creator which in turn invokes the Media Browser.

2. The Media Browser is opened and displays the Photo Media Gallery. Jack uses gestures, via the Gesture Interface, to browse through the pictures stored in the gallery and selects the picture he wants to use, a picture of a house with a garden.

3. As soon as the picture is selected the Media Editor is activated and the appropriate Media Controls are displayed. Jack wishes to adjust the brightness and the contrast of the picture, to make it seem as if it is dusk. He gazes at the brightness control icon, the Gaze Interface interprets this and then, with gestures, Jack adjusts the value of the picture's brightness and the effect is displayed back to him. When he is happy he uses an appropriate gesture to end the picture brightness editing.

4. Jack then gazes at the contrast control icon and this time, with gestures, he adjusts the value of the Picture contrast. Again, when he is satisfied with the result he uses a gesture to indicate that he has finished editing the pictures contrast.

5. Jack is now satisfied with the picture's properties and so uses a gesture to indicate this. The Media Controls disappear from the stage. Jack then uses gestures to position the photo within the central stage. These gestures are interpreted by the Gaze Interface and handled by the Presentation Creator

6. Jack then proceeds to add a sound clip to the presentation, using similar steps to those described above. However, this time he alters the pitch and volume properties of the media.

4.5.2 Displaying and saving the presentation

In this scenario Jack uses the Presentation Conductor system to display the presentation he has created and then to save it.

Steps:

1. Jack decides he wants to view the presentation from the beginning. From the Presentation Creator Module he invokes the Presentation Player with the use of the appropriate gesture. The picture is displayed and the sound is played according to the modifications that Jack did previously. With a gesture Jack plays the presentation again. He wishes to examine part of the sound clip. With the appropriate gesture he navigates to that position and plays the presentation from that point on. When he has finished viewing the presentation He indicates this with the appropriate gesture and the Presentation Player closes.

2. Jack wishes to save his Presentation. With the appropriate gesture the Presentation Creator stores the Presentation Definition of the currently edited presentation in the Presentations Media Gallery of the Media Repository.

4.5.3 Browsing, loading, and displaying an existing presentation

In this scenario Jack uses the Presentation Conductor system to browse through existing presentation definitions, loading one and finally displaying it.

Steps:

1. Jack wants to view a Presentation Definition from the Presentation Gallery. Using the same approach as described in section 4.5.1. Jack browses the Presentation Media Gallery. He previews the stored Presentation Definitions and with the appropriate gesture selects the one he wants to view.

2. Upon selection the Presentation Definition is loaded and the presentation is re-constructed by the Presentation Creator from the relevant Media Objects according to the prescribed order and state.

3. Jack using a similar approach to that described in section 4.5.2 then views the Presentation that has been loaded. Using gestures he starts and stops the presentation from different points.

4.6 Chapter summary

This chapter has presented the Conductor Interaction Method that is proposed by this thesis.

The interaction method itself is based around two novel metaphors that have been developed, the Orchestra and Conductor metaphors. These metaphors play a fundamental role within the Conductor Interaction Method and allow the user to interact with the computer via the use of gaze and gesture interactions. The method makes use of two-phased interaction, in which gaze is used to select and gestures are used to manipulate. This combination means the Midas Touch problem can be avoided, whilst complex interactions can still take place.

In order to demonstrate the Conductor Interaction Method, this thesis also describes an system that utilises the method. This chapter has presented the architecture for the Presentation Conductor, and in the next chapter the realisation of this architecture is presented. Finally, to help illustrate how the Conductor method and the Presentation Conductor will work in practice, a set of scenarios have been provided.

The following chapter describes a developed implementation based on the Conductor Interaction Method and more specifically the Presentation Conductor architecture. This implementation seeks to realise the method that has been presented in this chapter. The evaluation of this method is presented in Chapter 6.

Chapter 5 – The Presentation Conductor: an Implementation of the Conductor Interaction Method

5.1 Introduction

This thesis justifies, and describes the implementation of, a novel, more natural, Human Computer Interaction method, the Conductor Interaction Method, which makes use of gestures and user gaze as an input.

In chapter 4 the Conductor Interaction Method was discussed in detail. A brief discussion of the use of metaphors within HCI was first provided, before presenting the two novel metaphors that form the basis of the method, the Orchestra and the Conductor metaphor. The Orchestra metaphor represents the environment in which users interact. The Conductor metaphor represents the ways in which the user interacts with this environment.

Chapter 4 also presented the architecture for the Presentation Conductor system, a prototype multi-media presentation creation system developed to realise, demonstrate and evaluate the Conductor Interaction Method. In order to help visualise the use of the Conductor Interaction Method and the Presentation Conductor a series of scenarios were also provided.

In this chapter, the implementation of the Conductor Interaction Method, and the Presentation Conductor system that utilises it, are presented. The chapter is split into two main sections. The first examines how the Conductor Interaction Method has been implemented. It describes the implementations of the Conductor and Orchestra metaphors, and also discusses how two-phased interaction has been supported, as well as the efforts that have been made to reduce the user's learning overhead.

The second part of the chapter focuses on the implementation of the Presentation Conductor. It discusses how the Conductor Interaction Method has been incorporated within the system, and how its functionality has been provided for. Examples are provided throughout to assist in understanding, and also to suggest what the tool is like to use.

The evaluation of the Presentation Conductor and the Conductor Interaction Method is presented in Chapter 6.

5.2 Implementing the Conductor Interaction Method

The Conductor method aims to provide users with a more natural way of interacting with a computer, by featuring both gaze and gesture user interaction. As discussed in chapter 4, in order to help the user understand and use the interface, the Conductor Interaction Method embodies two metaphors, the Orchestra and Conductor metaphors.

To realise the Conductor Interaction Method the Presentation Conductor system has been developed, based on the architecture presented in chapter 4. Not only does the Presentation Conductor system enable the demonstration of the Conductor Interaction Method, but also allows for it to be evaluated.

To re-iterate, the Conductor Interaction Method and its implementation within the Presentation Conductor system seeks to provide:

- A more natural interface that utilises gaze and gestures but is nevertheless capable of supporting sophisticated activities.
- An interface that uses the Orchestra and Conductor metaphors.
- A two-phased interaction method. This method also overcomes the Midas Touch problem.
- Significantly reduced learning overhead compared to existing interaction methods.

The following sections discuss how the Conductor method has been implemented for use within the Presentation Conductor system.

5.2.1 Implementing the Conductor Metaphor

As has been previously described, the Conductor metaphor represents the user and the way he/she can interact with the application and involves both gestural and gaze inputs. The implementation of the Conductor metaphor tackles each of these inputs individually and each is described below.

The capturing of gestures within the implementation of the Conductor metaphor is performed using the process illustrated in Figure 5.1.

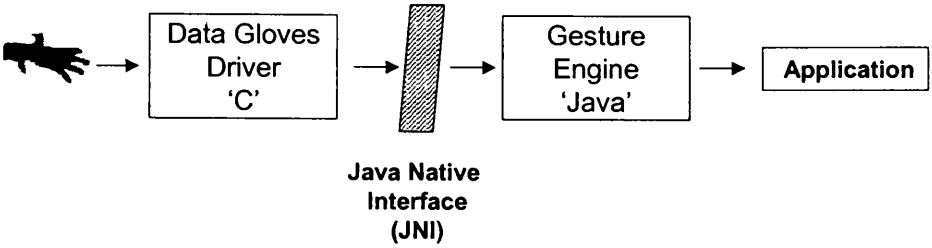


Figure 5. 1 - Capturing and recognising gestures within the Conductor metaphor

Data Gloves – in order to capture the user's gestures 5DT optical fibre based gloves are used (5DT). As discussed in chapter 2, these gloves have good precision and are suitable for tasks that do not require very high precision of identification of hand movement. They are also modestly priced.

Data Gloves Driver – data from the gloves is handled by a Windows driver that is written in C and is supplied with the gloves.

Java Native Interface – because the Gesture Engine is written in Java it needs to interface with the C based data glove driver via the Java Native Interface.

Gesture Engine – the Gesture Engine recognises the gestures that are being performed based on the data being output by the data gloves driver, and also handles the mapping of these gestures to keystrokes. The implemented Gesture Engine can recognise a range of gestures as described in table 5.1. Currently the gestures are hardwired into the Gesture Engine via coding. However the addition of new gestures is relatively easy, as the data glove driver produces a nine-character string that represents all the characteristics of the gesture, i.e. hand, position of each of the five fingers, roll and pitch. This nine-character string is then used to represent a gesture. In future developments of the gesture engine a more user-friendly method of adding and editing gestures can be incorporated into the gesture engine.

Once a gesture has been recognised it is then mapped to a keystroke. Mapping gestures to keystrokes provides a number of advantages, in particular keystrokes are an easy input for programming languages to handle, and also are easy to test during application development. The Gesture Engine, itself, is written in Java and provides a generic interface that means it can be easily incorporated into applications other than the Presentation Conductor.

Application – the identified gestures, represented as keystrokes, are then fed into the application as an input.

The capturing of gaze within the implementation of the Conductor metaphor is performed using the process illustrated in figure 5.2.

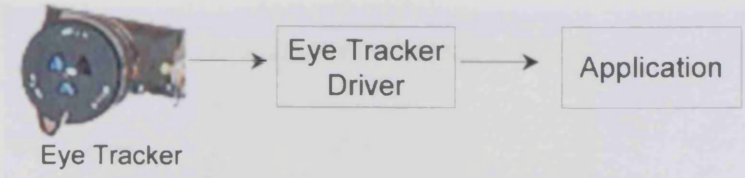


Figure 5. 2 - Capturing and recognising user gaze within the Conductor metaphor

Eye Tracker – in order to capture the user's gaze a LC Technologies VOG (LC Technologies Inc.) eye tracker is used. As discussed in chapter 2 and Appendix B, this type of eye tracker works by measuring the relative positions of the glint and the bright eye produced when infrared light is shone into the eye. This type of eye tracker was used because it is considered the most comfortable and less intrusive for the user.

Eye Tracker Driver – data from the eye tracker is handled by a Windows driver that is supplied with the eye tracker. The driver interfaces with Windows and moves the native mouse pointer. Unlike with the gesture capturing, no further processing is required to integrate this interaction method with the prototype.

Application – the user's gaze, tracked and represented as mouse movements, is input into the application.

Using these two methods makes it possible to realise the Conductor metaphor within the Presentation Conductor system. Its implementation, however, is also generic enough to allow it be utilised with other suitable applications.

5.2.2 Implementing the Orchestra Metaphor

As discussed in chapter 4, the Orchestra Metaphor represents the interaction environment for the user. The purpose of the metaphor is to graphically present to the user the available resources that are available for him/her to manipulate. The Conductor metaphor is then used to interact with the environment.

The Orchestra Metaphor itself is closely linked to the application that it is to be used with, and consequently this influences its implementation. As the Presentation Conductor system is being used to realise the Conductor method, the implementation of the Orchestra metaphor, presented here, takes this into account.

Figure 5.3 illustrates the realisation of the Orchestra Metaphor that has been developed for the Presentation Conductor system.

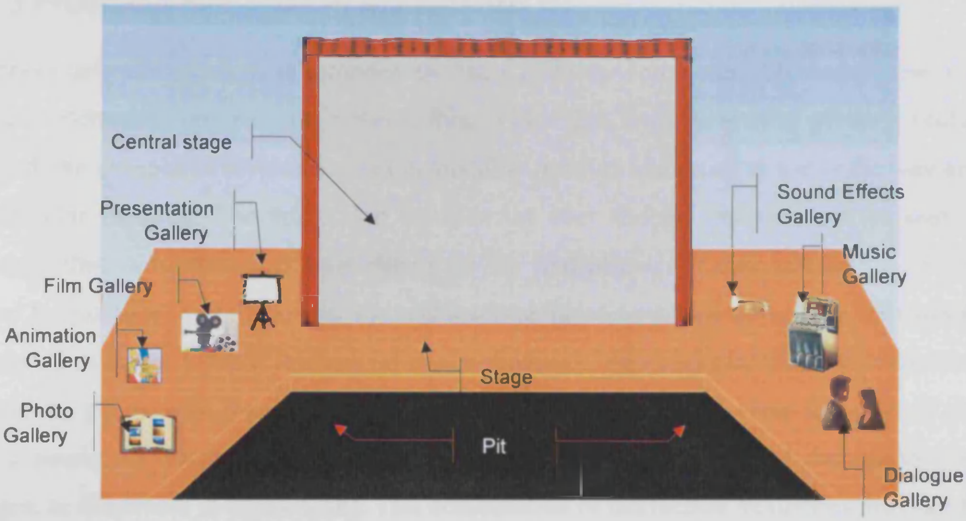


Figure 5. 3 - The implementation of the Orchestra metaphor used with the Presentation Conductor

The implementation of the Orchestra metaphor features a stage and central stage as discussed in chapter 4. The stage is the part of the graphical interface where the objects that the user interacts with are placed, while the central stage is the part of the graphical interface in which the results of the interaction are displayed, i.e. the edited elements of the presentation.

The range of "instruments", i.e. *media objects,* that are presented on the stage, are:

- *A Music Gallery (Jukebox)*: This represents various sound objects and includes music and environmental sounds.
- *A Sound Effects Gallery (Trumpet)*: This represents a series of sounds that the user can use.
- *A Dialogue Gallery (Two people talking):* This represents a series of short phrases that the user can use.
- *A Film Gallery (film camera):* This represents a series of films (i.e. digitised video sequences) that the user can use.
- *An Animation Gallery (Cartoon characters):* This represents a set of animations that the user can use.
- *A Photo Gallery (Picture book):* This represents a series of pictures that the user can use.
- *A Presentation Gallery (Projection Screen):* This represents previously created and stored presentations.

Although only the above galleries have been implemented in the Presentation Conductor, additional media types could also be used to extend the system at a later date.

5.2.3 Providing two-phased interaction

As previously discussed, it is intended for the Conductor Interaction Method to use a two-phased interaction process. To achieve this, within the implementation of the Conductor method, the interaction process uses each modality (gesture and gaze) in a specific way and in a particular function. The interaction between the user and the interface can be seen as a *dialogue* that is comprised of two phases. In the first phase, the user selects the on-screen object by *gazing* at it (for example, the user looks at the *picture book* object and activates it, as illustrated in figure 5.4a). If the user looks elsewhere the object is automatically deselected. In the second phase, with the *gesture* interface the user is able to manipulate the selected object (for example, by flexing vertically his /her right hand the user can scroll through the list of images, as illustrated in figure 5.4b). This combination of interaction modalities provides two-phased interaction. This approach means that the Midas Touch problem can be overcome.

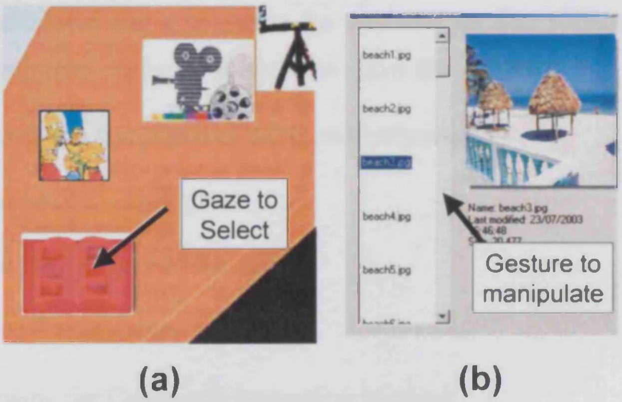


Figure 5. 4 – How two-phased interaction has been implemented

5.2.4 Assist in reducing the learning overhead

It is also desired for the Conductor Interaction Method to reduce the learning overhead required by the user to utilise the interaction style and any respective applications. For the implementation of the method and the Presentation Conductor system a small number of gestures are used. These gestures correspond to gestures that the user would normally use either in performing a task or in representing the task in question when describing it in a pantomimic way (for example, turning a knob). The gestures used are classified as emblems or pantomimes on Kendon's Continuum of gestures (Mc Neil, 1992), discussed in Chapter 2. By using gestures that are mostly pantomimes it is intended that both cross-cultural misinterpretations and the learning overhead will be reduced.

The gesture vocabulary that is used within the Presentation Conductor is presented later on in this chapter (in Table 5.1)

5.3 Implementing the Presentation Conductor Application

This section presents the implementation of the Presentation Conductor application. An overview of the implementation is first provided, before the key functionality is discussed in detail.

5.3.1 Overview

As previously stated, the purpose of the Presentation Conductor system is to provide a realisation of the Conductor Interaction Method and the Presentation Conductor architecture that have been presented in chapter 4. Not only will it enable the Conductor Method to be demonstrated, but it will also facilitate its evaluation.

The Presentation Conductor itself is a tool that enables the user to create and display multimedia presentations. These presentations can be comprised of a variety of multimedia objects that can be modified and arranged in the way that the user finds suitable. The main features of the Presentation Conductor application allow the user to:

- Create a presentation composed of multimedia objects
- Manipulate the multimedia objects
- Store created Multimedia presentations
- Display created presentations
- Achieve this using the Conductor Interaction Method

The main application is based on the architecture presented in chapter 4 and has been developed using Visual Basic.NET (Microsoft Corp.). This development tool was chosen for its ease in building applications that have a complex Graphical User Interface (GUI). The ability to maintain the control of the application's GUI at all points of the interaction is essential in this application, mainly due to the fact that the user interacts using two different modalities. A further reason for using VB.NET was its ease in handling events from the eye tracker.

The Media Repository used by the application (to store the Media Objects, Media Galleries, etc) is represented by the local file system.

To help present the Presentation Conductor application, its description is broken down into the following topics:

- Incorporating the Conductor Interaction Method within the application
- Supporting the creation and editing of a presentation

- Supporting the playing, saving and loading of a presentation

Real examples are also provided to illustrate the features in use.

5.3.2 Incorporating the Conductor Interaction Method within the application

As discussed previously, the Conductor method utilises both gesture and gaze inputs. The gestures that the user performs are mapped onto key presses, which the Presentation Conductor maps to commands that invoke specific functionality. The implementation of the Conductor metaphor that has been presented in this chapter allows for a number of gestures to be used and recognised. Table 5.1 illustrates how each of these gestures is mapped to a specific functionality within the Presentation Conductor appliction. These gestures were derived from studying how users perform similar tasks (using similar physical objects) in the real world, and based on gestures commonly used in the western society to indicate direction. For example, adjusting the volume of an amplifier by turning the volume control knob, or how a policeman stops traffic by using a flat hand gesture.

Gesture	Name	Function
	***Left hand twist*** All fingers are flexed Hand performs a twisting act from right to left and vice versa	Used to adjust properties of a selected media object, by turning a knob.
	***Right hand twist out*** All fingers are flexed Hand in a fist performs a twisting action from left to right only	Used to play a presentation
	***Right hand Point*** Index and thumb stretched, all other fingers are flexed No movement	Used to select an item from a list.
	***Left hand Point-Down*** Index and thumb stretched, all other fingers are flexed. Starting position is pointing up, hand twist to the right and ends pointing downwards	Used to delete a selected item and send t to the pit

Table 5. 1 Mapping between gestures and operations within the Presentation Conductor

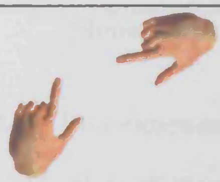
Gesture	Name	Function
	***Right hand Scroll Up & Down*** All fingers are stretched Hand performs a vertical flex movement	Used to scroll through the contents of the active Media Gallery.
	***Right hand Flex and hold Up & Down*** All fingers are stretched Hand performs a vertical flex movement, and pauses at the extreme high or low	Used to adjust vertical position or height of a displayable media object
	***Left hand "Stop"*** All fingers are stretched Hand raised flat in front of the user	Used to stop playback of a Media object or Presentation
	***Left hand vertical flat*** All fingers are stretched Hand performs a horizontal movement	Used to adjust horizontal position to the right of a displayable media object
	***Right hand vertical flat*** All fingers are stretched Hand performs a horizontal movement	Used to adjust horizontal position to the left of a displayable media object
	***Hands Framing*** Index and thumb stretched, all other fingers are flexed Both hands forming an index point, placed index to index	Used to initiate proportional size modification for displayable media objects
	***Both hands vertical flat, palms facing*** All fingers are stretched Both hands stretched, vertical to the ground, palms facing each other	Used to proportionally decrease the size of a displayable object
	***Both hands vertical flat, palms out*** All fingers are stretched Both hands stretched, vertical to the ground, palms facing out	Used to proportionally increase the size of a displayable object

Table 5.1 (cont.) Mapping between gestures and operations within the Presentation Conductor

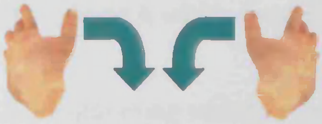
Gesture	Name	Function
	*Both hands flat* All fingers are stretched Both hands flat, parallel to the ground	Used to finalize an operation.
	*Both hands "shut the box"* All fingers are stretched Both hands stretched, vertical to the ground, palms facing each other, do an inwards movement, as if shutting a box	Used to Exit from the application

Table 5.1 (cont.) Mapping between gestures and operations within the Presentation Conductor

The Conductor metaphor also translates a user's gaze into PC mouse movements. The Presentation Conductor can interpret these to signify certain events, for example looking at a Media Gallery on the stage. Table 5.2 illustrates the simulated mouse events that are handled by the Presentation Conductor.

Native Mouse Pointer Events	Description
MouseEnter	Used when the User looks at a selectable object
MouseLeave	Used when the User looks away from a selectable object
MouseMove	Used when the User looks at a new position on the screen

Table 5. 2 - How mouse events are mapped to functionality within the Presentation Conductor

The combination of these two interaction methods allows the user to fully utilise the Presentation Conductor application.

The actual process of presentation creation itself is carried out in the environment that is represented by the Orchestra metaphor. The Media Galleries on the stage represent the resources that can be used within the presentation, and the central stage is where the presentation is actually constructed.

5.3.3 Supporting the creation and editing of presentations

The chief function of the Presentation Conductor application is to allow users to create and edit multimedia presentations.

Within the tool, a presentation represents a sequence of Media Objects that are to be displayed. The position of these Media Objects within the visual display can be set (i.e. their x and y co-ordinates), as well as the alteration of some of their properties (for example, a picture's brightness). Created presentations can then be saved as Presentation Definitions (c.f. chapter 4).

The actual creation and editing of a presentation can be broken down into three stages. Firstly, browsing and selecting the appropriate Media Object, secondly, editing its properties and finally, placing it within the presentation.

5.3.3.1 Supporting user browsing of the Media Galleries

To provide the user with the resources with which to construct their multimedia presentation, the Presentation Conductor makes use of a number of Media Galleries that are presented on the stage as part of the Orchestra metaphor (as shown in figure 5.3).

The process of browsing Media Objects and selecting one involves input from the user via both gaze and gestures. To commence the activity the user first needs to use the gaze interface to select the relevant Media Gallery. As discussed earlier in the chapter, the users gaze is converted into a mouse event that is then processed by the Presentation Conductor. If the user gazes on a Media Gallery icon it becomes highlighted in red (as shown in figure 5.5). If the user looks elsewhere (for example, at an empty part of the stage), the gallery becomes un-highlighted. As a result of this (and also helped by the fact that gestures are needed to complete the browsing activity) the Midas Touch problem is avoided.

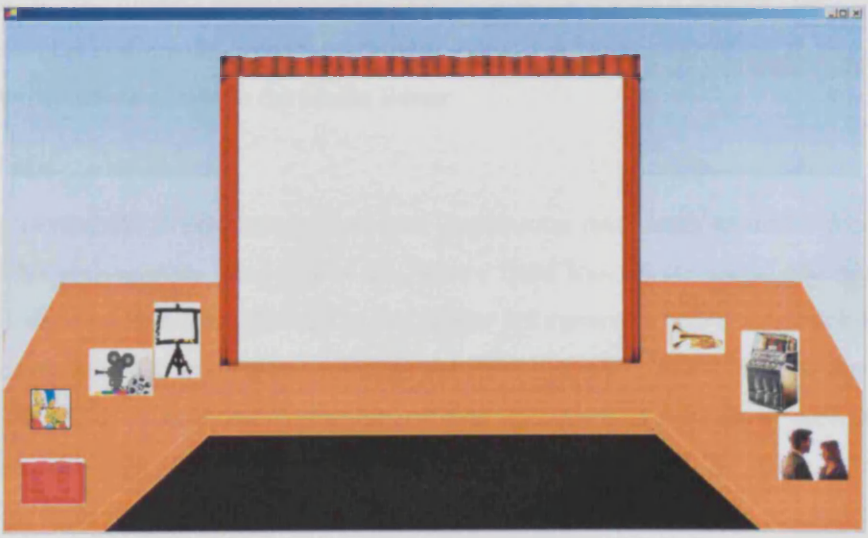


Figure 5. 5 - Using gaze to select a Media Gallery

With the use of the appropriate gesture (***Gesture: right hand Point***) the user confirms the selection of the Media Gallery, which results in the gallery becoming activated and the Media Browser being displayed. When this happens all the galleries except for the selected one fade, so that it is clear to the user which gallery is being browsed (as shown in figure 5.6). The

Media Browser displays all the Media Objects that exist within that gallery as a list that the user can navigate through with the use of gestures (***Gesture: right hand scroll up & down)***. A representation of the currently selected Media Object is displayed so that the user can preview it before deciding whether or not to use it. For an image, an animation and a video, a thumbnail representation is used (as illustrated in figure 5.6). For an audio-based object, the currently selected object is played.

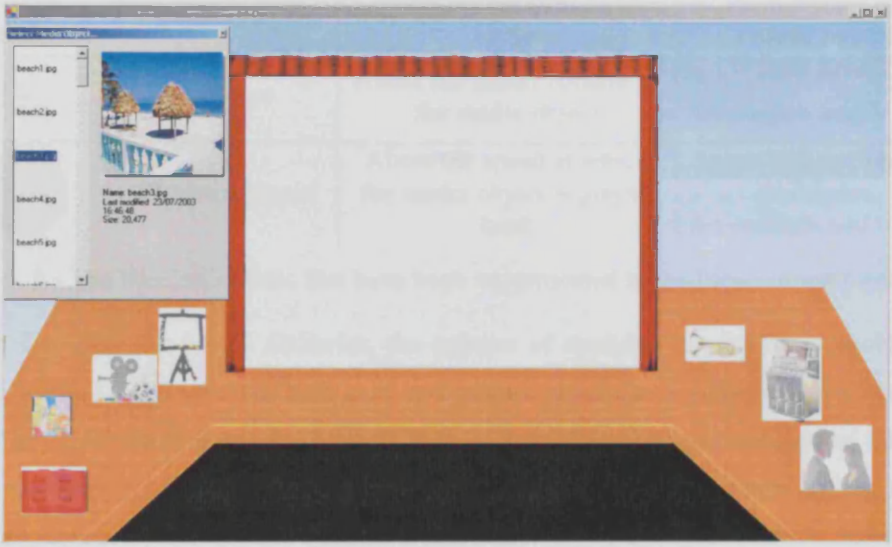


Figure 5. 6 - Using gestures to browse a Media Gallery

When the desired Media Object has been located, with the use of a gesture (***Gesture: Right hand Point***) the user can indicate that they wish to use it in their presentation. At this point the Media Object is then passed to the Media Editor.

Example

Jack has started the Presentation Conductor Application and wants to add a picture of a beach to his presentation. He gazes at the Picture Book icon on the stage. The application interprets this and highlights the icon in red (lower left corner in figure 5.5). Jack continues to stare at the icon and in doing so activates the Media Browser. In the browser Jack scrolls through the list of pictures by flexing his palm up and down. A preview of each picture is shown and Jack manages to find a suitable beach picture (figure 5.6). He points at it with the index of his right hand and the picture is loaded into the Media Editor.

5.3.3.2 Editing the Media Objects

Once a Media Object has been selected, the user is given the opportunity to alter some of the properties that it may possess. Within the developed version of the Presentation Conductor that is presented here, four types of Media Controls have been provided. Table 5.3 summarises these and also highlights which types of Media Object they can be used with. It would not be difficult to add additional Media Controls to the Presentation Conductor if so desired.

Control Icon	Control Name	Control Description	Media Objects it Manipulates
	Brightness	Alters the brightness of an Image	Images from the Picture gallery
	Contrast	Alters the contrast of an Image	Images from the Picture gallery
	Volume	Alters the sound volume of the media object	Sound effects, Music. Dialogues, and Video
	Playback Speed	Alters the speed at which the media object is played back	Sound effects, Music, Dialogues, Animations and Video

Table 5. 3 - The Media Controls that have been implemented in the Presentation Conductor

As with browsing the Media Galleries, the process of applying the Media controls to the selected Media Object involves both gaze and gesture input. Once a Media Object has been selected in the Media Browser, the Media Editor is launched and a representation of the object is displayed within. In the case of an image, an animation and a video media object, again, a thumbnail representation is used. This serves as a preview of the modifications that the user applies to the Media Object using the Media Controls. A generic icon is used to represent audio-based objects. When audio based Media Controls are used, any modifications are played immediately after each modification is applied. Figure 5.7 illustrates the Media Editor and the two Media Controls for manipulating an image.

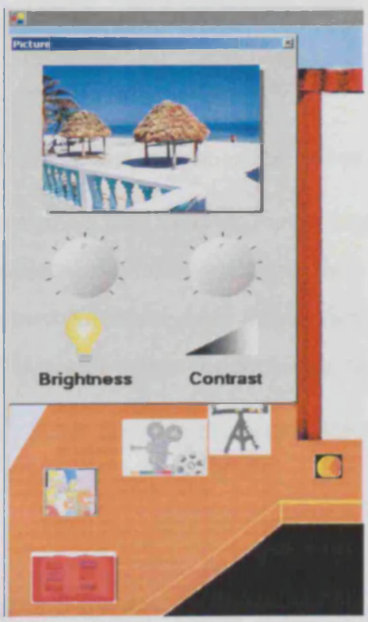


Figure 5. 7 – The Media Editor

To commence a modification the user first needs to gaze at the Media Control that they wish to use. Upon gazing at a Media Control, its control knob becomes highlighted in yellow (as illustrated in figure 5.8) and the control becomes activated. At this point in the interaction, if the user gazes away from the control, the Media Control returns to its inactive state and is displayed in grey.

While a control is active the user may use the appropriate gesture (***Gesture: left hand twist***) to modify the respective properties of the Media Object. From this point until the interaction is complete or cancelled, even if the user gazes at another location on the display, the Media Control will remain active. As the gesture is performed, the control knob turns and the preview of the Media Object is updated. This enables the user to see or hear the effect they are having on the Media Object. Once the modification has been completed another gesture (***Gesture: both hands flat***) is used to end the editing of the Media Object with that Media Control. Figure 5.8 illustrates how the Media Editor can be used to manipulate the properties of a Media Object.

Figure 5. 8 - Using the Media Editor to manipulate the properties of a Media Object

Each Media Control may be selected and used as many times as the user finds necessary. When the user has finally completed the editing activity they can leave the Media Editor by performing the appropriate gesture (***Gesture: both hands flat***). The edited Media Object can then be placed on the Central Stage and within the presentation.

Media Objects that have already been placed in the presentation can be re-edited whenever the user wishes.

Example

Having picked the beach image, it is displayed within the Media Editor. There are two Media Controls available to Jack, brightness and contrast (figure 5.7). Jack decides he would like to *darken the picture slightly. He gazes at the brightness icon. The application recognises this and highlights the brightness control knob in yellow. Jack then twists his left hand and in doing so, adjusts the brightness of the picture (figure 5.8). When Jack is satisfied he holds both his hands out flat and exits the Media Editor.*

5.3.3.3 Placing, scheduling and deleting Media Objects

Once the user has indicated that the editing of the Media object is complete, the Media Editor is closed and the object appears on the Central Stage. Audio objects are depicted on the stage by a generic visual representation (a loudspeaker). Visual objects such as images, animations, and videos are placed in the centre of the central stage, and are sized to fit the stage (with video and animation media objects, the first frame is displayed to represent the object).

Placed visual Media Objects can have their height, width and position on the central stage adjusted by the appropriate sequence of gestures. These are summarised in table 5.4. By default the 'adjust position' action is activated and so no gesture is required to initiate or to complete this action. When the user is satisfied, they can exit the adjustment of the visual Media Object activity with the appropriate gesture (***Gesture: both hands flat***).

Desired Action	Initiation of Action	Attribute Change	Completion of Action
Adjust Height	***Gesture: Hands horizontal facing***	Increase Height – ***Gesture: Right hand "Top"*** Decrease Height – ***Gesture: Left hand "Bottom"***	***Gesture: both hands flat***
Adjust Width	***Gesture: Both hands Vertical***	Increase Width – ***Gesture: Right hand "Flex"*** Decrease Width – ***Gesture: Left hand "Flex"***	***Gesture: both hands flat***
Adjust Position	Automatic (default action)	Move Up – ***Gesture: Right hand "Top"*** Move Down – ***Gesture: Left hand "Bottom"*** Move Left – ***Gesture: Left hand "Flex"*** Move Right – ***Gesture: Right hand "Flex"***	Automatic (default action)

Table 5. 4 - Adjusting the attributes of visual Media Objects on the Central Stage

The schedule of the presentation is represented by a row of footlights that appear at the foot of the central stage (as illustrated in figure 5.9). Within a presentation one footlight represents a *scene,* where a scene can be made up of a number of Media Objects (for example, an image and a sound).

When a Media Object is placed on the Central Stage, by default it is scheduled to be part of the currently selected scene, which is indicated by the lit (yellow) footlight. During the presentation the Media Object will remain on the stage until the end of the scene. The Media Object with the longest duration determines the time length of a scene. Media Objects such as sound and video clips are played for their entire duration. For Media Objects, such as images, that have no duration associated with them, a default duration of 3 seconds is given. This is used in the event that such a Media object is the only object of a scene.

For example, a scene can contain an image, a sound with a 10 second duration and an animation that runs for 7 seconds, giving the scene a total duration of 10 seconds. The image will be displayed for the whole of these 10 seconds.

It is also possible to edit the other Media Objects within a scene, by selecting the relevant object. This is achieved by the user gazing at the relevant Media Object, and then using the appropriate gesture to select it (***Gesture: right hand point***). The user is then able to re-position and adjust the media object as described above. It is also possible for Media Objects to be deleted from a scene by selecting the appropriate object with the use of gaze and gesture, in the same way as when editing, and then using the appropriate gesture (***Gesture: Left hand point down***) to mimic directing it into the pit. This Media Object is then removed from the scene.

To change the currently selected scene within the presentation schedule, an alternative footlight needs to be selected. This is achieved by the selecting the footlight using the same process as for selecting a Media Object within a scene. Once this is done, the user is able to edit the Media Objects that comprise the scene, as described above.

In order to create a new scene within the presentation schedule the user performs the appropriate gesture ***(Gesture: Left hand "Stop")*** and this results in adding a new scene (and footlight) at the end of the schedule.

Figure 5. 9 – Placing a Media Object on the Central Stage

Example

The beach image that Jack edited is now displayed on the Central Stage. The image is automatically centred on the display area and covers it completely. Jack may now adjust the size of the image and its position on the central stage. Jack decides he would like to shrink the picture slightly. He positions his hands horizontally, palms facing each other, in front of him to initiate the adjustment of the images height. He then flexes his left hand that is facing up, mimicking a "push" to the image. The image height decreases. Jack is satisfied with the height of the image and holds both his hands out flat to end the height adjustment action.

He now positions his hands vertically flat in front of him, palms facing each other, to initiate the adjustment of the image width. He flexes the left hand towards the centre of the image, again mimicking a "push" to the image, to reduce it width. The image width decreases and Jack is satisfied. Again he holds both his hands out flat to indicate he has completed the width adjustment. The image is no longer centred on the central stage. Jack places his right hand flat in front of him and flexes it mimicking a "push" down to the image. He then moves his right hand to a flat and vertical position and flexes it mimicking a "push" to the left.

When Jack is satisfied with the adjustment of the image he holds both his hands out flat to set the adjustments.

The process of selecting Media Objects, adjusting their properties, positioning them on the central stage and within the presentation schedule, can be repeated until a presentation has been fully created.

5.3.4 Supporting the playing, saving and loading of a presentation

As well as supporting the creation of presentations, the Presentation Conductor also allows for them to be played, stored, and reloaded. These three functions are discussed below.

5.3.4.1 Playing a presentation

At any point when a presentation is not being edited it is possible to play back the presentation using the appropriate gesture ***(Gesture: Right hand twist out).*** Upon doing this the presentation will start playing in the central stage (as show in figure 5.10). The presentation will continue to play until it reaches the end, or until it is interrupted by the user making the appropriate gesture ***(Gesture: Left hand "Stop").*** A stopped presentation can be either resumed by again performing the 'play presentation' gesture ***(Gesture: Right hand twist out)*** or completely abandoned, with the finalisation gesture (***Gesture: both hands flat***).

Figure 5. 10 - Playing a presentation

Example

Jack has been creating a Presentation about Beach activities, where he has incorporated a variety of Media Objects from the various Galleries available to him. After incorporating an audio media object he decides to play the presentation to see how it all ties together. He stretches his right hand in front of him in a fist and twists it to the right. The presentation starts playing. At that point Jane walks into the room to ask him a question. Jack stops the presentation playing by raising his left hand flat in front of him. When Jane leaves he puts his right hand out in a fist again and turns it to the right and the presentation continues playing, up to the end.

5.3.4.2 Saving a presentation

Whenever a user embarks on creating a new presentation, prior to using the Conductor Interaction Method interface, they are asked to type in a name for the presentation. This is then used to create a Presentation Definition file (see chapter 4). Throughout the presentation creation process a record of the used media objects, the modifications that have been applied to them, and their order within the presentation schedule, is stored within this Presentation Definition file. This file is closed when the application stops running.

The user indicates their desire to end the presentation creation process and to exit the application with a sequence of gestures. The first gesture indicates that they have finalised the process of creating the presentation ***(Gesture: Both hands flat)***. The second gesture indicates their desire to exit from the application ***(Gesture: Both hands flat twist out)***. At this point the Presentation Definition is stored to disk and can be accessed again at a later date if needs be.

Example

Jack is happy with the presentation and wants to leave it as it is and exit. He indicates that he has finished all processes he was doing by holding both his hands out flat. He now indicates that he wants to exit the application by twisting his hands so that the palms are facing up. The presentation is stored to the previously given name and the application closes. Jack can now remove the gloves.

5.3.4.3 Loading a presentation

The process of selecting and loading a presentation is identical to that of selecting a Media Object from a Media Gallery (Presentation Definition files being stored in Media Galleries). As previously discussed, the user first uses the gaze interface to select the relevant gallery, which in this case is the Presentation Gallery. Once this is selected the user can browse the Presentation Definitions kept in the gallery in the Media Browser (as illustrated in Figure 5.11) in the same way that they would browse the Media Galleries during the creation of a presentation.

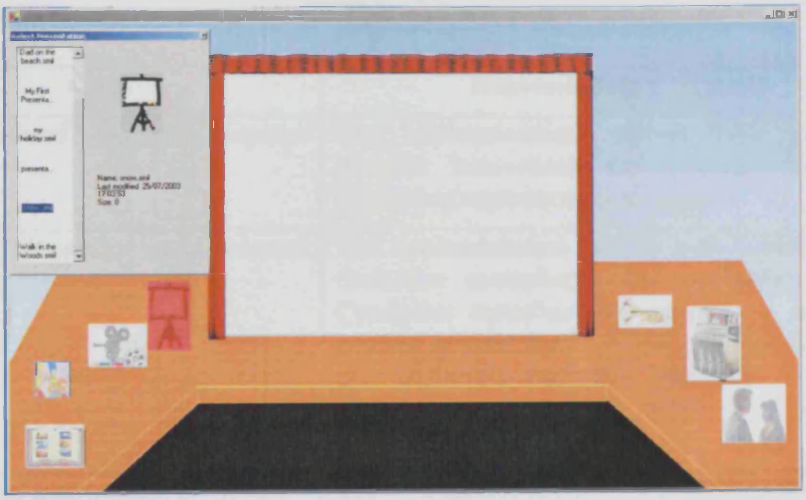


Figure 5. 11 - Browsing and selecting a Presentation Definition

Once the presentation is selected, it is loaded and appears in the central stage (as illustrated in figure 5.12) and can be played and edited in the ways previously described.

Figure 5. 12 - A loaded presentation

5.4 Chapter Summary

This chapter has discussed an implementation of the Conductor Interaction Method that is presented in this thesis, and demonstrated its use through the prototypical Presentation Conductor system. The Presentation Conductor system realises the architecture that was described in chapter 4.

Table 5.5 provides a breakdown of how the implementation of the Conductor Interaction Method has set out to satisfy the design goals that were specified in chapter 4.

Goal	Implementation Support
More natural Human-Computer Interface	The implementation utilises both gaze and gestures interactions and also is capable of supporting sophisticated activities
Use of the Conductor and Orchestra metaphors	The implementation utilises both Conductor and Orchestra metaphors. The realisation of the Conductor metaphor can interpret a significant number of user hand gestures. The realisation of the Orchestra metaphor provides a suitable environment for the creation of multi-media presentations.
Two-phased interaction	The implementation supports two-phased interaction, where gaze is first used to select, and then gestures used to manipulate and control.
Reduced learning overhead	The implementation has attempted to reduce the learning overhead by using gestures that users are familiar with and relate to the task at hand.

Table 5. 5 - How the implementation of the Conductor Interaction Method meets the design goals

As discussed in chapter 4, the Presentation Conductor application that has been developed enables users to create multi-media presentations. The implementation of this application enables users to browse Media Galleries, select Media Objects, edit their properties and place them within the presentation. Created presentations can be played back, stored and loaded at a later date.

The following chapter describes an evaluation that has been performed on the Conductor Interaction Method through the Presentation Conductor. The evaluation aims to assess the usefulness of the method and its application.

Chapter 6 – Evaluating the Conductor Interaction Method through the Presentation Conductor

6.1 Introduction

This thesis has argued for the consideration of a novel, more natural, Human Computer Interaction method, the Conductor Interaction Method, which makes use of gestures and gaze as user input. The Conductor Interaction Method was discussed in detail in Chapter 4.

In Chapter 5 an implementation of the Conductor Interaction Method was presented in the form of the prototypical Presentation Conductor system. This tool realises the model and architecture previously described in chapter 4.

This chapter describes the evaluation that was performed using the prototypical Presentation Conductor system. The evaluation studies aimed to assess the usefulness of the Conductor Interaction Method, as well as to obtain feedback on its application. In particular, the studies intended to assess the implementation of the method and the learning overhead in using the method, in comparison with existing interaction approaches. Additionally, the appropriateness of the implemented application was examined, as well as the impact the underlying technology might have on the effectiveness of the method. As one of the aims of the method was to assist inexperienced users in their interaction with the computer, groups of users possessing differing levels of computer experience were used in the studies. A control group was also used.

Given the prototypical nature of the Presentation Conductor system, formative and qualitative evaluation approaches were primarily used for the assessment. The NASA Task Load Index (Hart, 1998) was also used to obtain to obtain supportive quantitative feedback relating to user workload.

The following section discusses the motivation for, and the objectives of, the evaluation in more detail. Based on these objectives the next section discusses the various evaluation methods and justifies the selection of a formative and qualitative approach for this evaluation. The structure of the evaluation is then discussed, and its two stages are described. The first stage focuses on obtaining feedback on the Conductor Interaction Method, while the second focuses on obtaining feedback on the potential effects of the chosen technology on the applicability of the Conductor Interaction Method. The chapter then provides an analysis of the results gathered from the two stages of the evaluation.

6.2 Motivation for the evaluation

Elliot Stern (Sommerland, 1992, p11) defines 'evaluation' to be:

"...any activity that throughout the planning and delivery of innovative programmes enables those involved to learn and make judgements about the starting assumptions, implementation process and outcomes of the innovation concerned."

Though Stern was providing this definition in the context of educational, social and organisational programmes, it is equally valid for the development of computer systems. The aim of the evaluation presented in this chapter was to assess the Conductor Interaction Method by considering its implemented form, i.e., the Presentation Conductor. The following key requirements of the evaluation were identified:

- To obtain feedback on the Conductor Interaction Method and thus assess the usefulness of the approach. In particular:
    - *If users find it easier to interact with computers using the Conductor Interaction Method than with existing interaction techniques*
    - *If the metaphors featuring in the method are useful and beneficial*
    - *If the learning overhead for the method is lower than with existing interaction techniques*
    - *If users favour such an approach over alternative approaches*

- Additionally, the evaluation aimed to obtain feedback on the application of the Conductor Interaction Method, in particular:
    - Feedback on the devices used
        - *If the technology used can influence the effectiveness of the Conductor Interaction Method, in particular gaze versus head pointing, and small screen versus large screen*
    - Feedback on the Presentation Conductor application
        - *If the Conductor Interaction Method is appropriate for this type of application, and whether users would prefer to use such an interaction approach for such an application*

For the latter major bullet point, because the Conductor Interaction Method relies on the use of two-phased multimodal interaction (gaze and gestures), it is also important to evaluate the devices and mechanisms that are used to capture the input from the user. This will not only provide insight into how users cope with the different interaction methods, but also whether the choice of input device can have an effect on the applicability and effectiveness of the method. Of particular interest was whether the use of gaze or head pointing and small or large displays could impact the effectiveness of the Conductor Interaction Method.

The rest of this chapter describes and discusses the evaluation that was carried out on the Conductor Interaction Method and Presentation Conductor system, and how it evaluates the issues mentioned above.

6.3 Evaluation approach

Evaluation techniques have been categorised into a number of methodologies, of which quantitative, qualitative, formative and summative are the most commonly used (Rammage, 1999). Table 6.1 provides a summary of these.

Name	Characteristics	Example
Quantitative	Typically uses the methods of laboratory science, involving strict and controlled experimentation to gather data that is statistically analysed and interpreted.	Measuring the frequency with which a user performs a right mouse button click.
Qualitative	Has its origin in social sciences. These types of evaluations tend to be informal studies and their feedback tends to be more subjective and descriptive. Analysis of the data can be more difficult but provides the evaluator with a better view of the user's opinion	Obtaining a user's opinion on how easy it was to learn to use the system to perform a specific task.
Formative	Is conducted during the development or improvement of a system, where the feedback is used to inform the future development of the system. The techniques used in this stage are usually qualitative, as the objective of this type of evaluation is information that will assist in the enhancement of the system.	Releasing beta versions of software to be assessed, prior to the full release.
Summative	Is conducted when the system is completed to assess whether the design objectives have been met. Usually quantitative methods are used and precise results are returned.	Testing a final product to check whether it complies to standards set by a user group or a standards organisation such as ISO

Table 6. 1 - Evaluation Techniques

When considering what evaluation approach to use for evaluating the Conductor Interaction Method and Presentation Conductor system, there were a number of influential factors, in particular:

- *The evaluation was to be performed on an HCI*

  It is difficult to generalise interface principles across users and domains. According to their background and domain of interest, users typically have different interface preferences, while interactions suitable for one domain might not be suitable for another. As a result this can make the use of quantitative techniques problematical. For the Conductor Interaction Method, such problems are exacerbated because the user is engaged in *multimodal* interaction. As the user interacts using more than one modality, it is of questionable benefit to quantify isolated interactions.

- *Most of the evaluation objectives are difficult to quantify*

  The evaluation focuses on the usability and usefulness of the Conductor Interaction Method. Such attributes are mainly subjective in nature and consequently difficult to quantify.

- *The Presentation Conductor system is a prototype*

  The Presentation Conductor system is a prototype system that has been developed to realise the Conductor Interaction Method. Because the system is not a complete product, formative rather than summative techniques would be more suitable for its evaluation. Boucherat (1991) has discussed the difficulties of performing evaluations in such circumstances.

As a consequence of these factors it was decided that a formative and qualitative approach would be most suitable for evaluating the Conductor Interaction Method and Presentation Conductor system. It was also decided that the NASA Task Load Index (Hart, 1987) would be used to help attempt to quantifiably analyse the qualitative results. The NASA Task Load Index can be used to measure the workload (based on six scales) that users perceive themselves to be under. A more detailed description of this method can be found in appendix D.

6.3.1 Evaluation Structure

In order to assess the issues that has have been identified earlier in this chapter, the evaluation focused on users making use of the Presentation Conductor to carry out a set of prescribed tasks. Three groups of users participated in the evaluation, a group of four experienced computer users, a group of three inexperienced computer users, and a control group that was also made up of three inexperienced computer users.

The experienced user group was made up of colleagues from the department who possessed extensive PowerPoint experience. The inexperienced users used within the evaluation included four first year Music students, two postgraduate Psychology students and a member of the security staff of Lancaster University, and although some had used computers for tasks specific to their discipline, email and browsing the Web, they had no or very little experience in using PowerPoint. The control group was used to identify whether the creation of a presentation with PowerPoint had any effect on how inexperienced users then interacted with the Presentation Conductor. The evaluation was carried out in two stages:

Stage 1: The first stage of the evaluation focused on obtaining general feedback on the Conductor Interaction Method, and in particular, how it compared with existing interaction techniques. The groups of experienced and inexperienced users were given a specific presentation to create by first using Microsoft's PowerPoint presentation package and then by using the Presentation Conductor system. Since the control group was used to identify whether the use of PowerPoint had any effect on interacting with the Presentation conductor, this group used only the Presentation Conductor to create the presentation.

Stage 2: The second stage of the evaluation focused on obtaining feedback on the use of technology with the Conductor Interaction Method. All three groups were again given the same presentation to create, but this time they had to use the Presentation Conductor system with a different set of devices.

Throughout both stages feedback was obtained via a number of mechanisms. The evaluation sessions were all videoed from two angles, one focusing on the display screen and the other focusing on the user. Verbal feedback was also obtained as a result of questions posed by the investigator and from the users 'thinking aloud' whilst performing the tasks. At the end of every stage the users were asked to fill in questionnaires (appendix E) and were also interviewed. The questions used in the questionnaires and the interviews were standardised for each stage and were identical for both groups. The questionnaires also included assessment activities related to the NASA Task Load Index.

6.4 Stage 1: Evaluating the method

This stage aimed to acquire feedback on the Conductor Interaction Method. In particular it focuses on making a comparison with other interaction techniques for creating presentations, namely Microsoft PowerPoint.

A semi structured evaluation approach was used where the users performed tasks and were encouraged to give verbal feedback.

6.4.1 Format of sessions

The set-up for this stage involved the user sitting in front of a workstation. An LC-Technologies eye tracker was mounted underneath the monitor so that the user's eye movements could be monitored. A small screen was placed to one side to assist the investigator in calibrating the eye tracker. A pair of 5th DT Data Gloves were connected to the workstation for use by the user during the stage. Figure 6.1 illustrates the set-up for this stage.

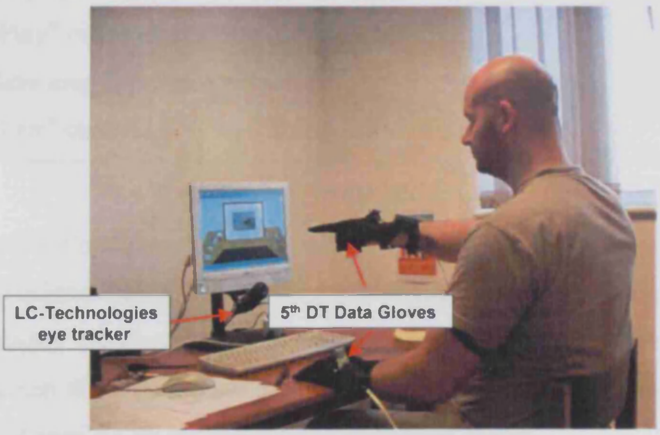


Figure 6. 1 - Stage 1 set-up

Each of the sessions in this stage consisted of six phases, with the exception of the sessions that involved the control group, which had only five phases (to take into account the fact that they did not create the presentation with PowerPoint). The six phases were as follows:

- *Preliminary briefing.* The session began with the investigator providing a brief overview of what the evaluation procedure would involve. The user was also provided with an introduction to computer-based presentations and how they can be beneficial.

- *PowerPoint Tasks (not performed by the control group).* The inexperienced or experienced user was then given a set of tasks to complete using Microsoft PowerPoint. The tasks that were provided are presented in Table 6.2. For the inexperienced users, prior to the tasks, the investigator provided a simple walkthrough of Microsoft PowerPoint that demonstrated its key features. When the tasks were completed the user was asked to complete the Task Load Index rating form for this set of tasks.

Task 1: Start a presentation
Launch the application from the desktop.
Task 2: Adding pictures to the presentation
Find a picture of a mountain to add to the presentation. Modify its brightness and contrast.
Task 3:Adding a sound to the presentation
Find a sound of birds to add to the presentation. Adjust the volume (and the speed) to your satisfaction.
Task 4: Adding an Animation
Find an animation of a person climbing a mountain. Position the animation in the display area of the presentation and make it occupy the lower left quarter portion of it..
Task 5 :Playing the presentation
Use the "Play" command to play the presentation
Task 6: Save and Exit the application
Use the "Exit" command to save and exit the presentation

Table 6. 2- Evaluation Tasks

- *Briefing on the Conductor Interaction Method and Presentation Conductor.* The user was then provided with a brief description of the Conductor Interaction Method and the Presentation Conductor system. Key concepts, including two-phased interaction were explained to them. Following this the user was presented with the application user guide (Appendix F) and given 5-10 minutes to read it.

- *System calibration and training.* Prior to making use of the Presentation Conductor, it was first necessary to calibrate the system to the individual user and to teach the gesture vocabulary. The user was asked to wear the data gloves and then to practice using the gestures that are used within the Presentation Conductor (c.f. chapter 5). The gesture recognition software was used to identify whether the gestures were performed successfully, and the investigator assisted when needed. For calibrating the eye tracker the user was asked to sit in a position that he/she could comfortably maintain for the rest of the evaluation stage.

- *Presentation Conductor demonstration.* The Presentation Conductor application was then launched and the investigator typed in the name for the user's presentation. The Presentation Conductor environment was activated and the investigator guided the user through the environment and how to perform the various operations within the application.

- *Presentation Conductor Tasks.* The user was then asked to complete the same set of tasks that they performed using PowerPoint (users from the control group were asked to perform the same set of tasks, although they had not previously performed them with PowerPoint). In order to ensure that the user's head movements were kept to a minimum (so not to effect the eye tracker calibration), the investigator read out the task instructions. Upon completion of the tasks the user was asked to complete the Task Load Index rating and the first part of the evaluation questionnaire.

The same evaluation structure was used for all users (except for members of the control group, as previously stated) and was filmed throughout from two angles.

Throughout the evaluation stage users were encouraged to voice any comments or problems they were having by responding to questions posed by the investigator at certain points within the stage. The question guide used by the investigator is shown in Appendix G.

After completion of this stage the users were allowed a five minute break before proceeding to the second, final, stage of the evaluation

6.4.2 Discussion of sessions

This section provides a discussion of stage 1 of the evaluation in terms of the following:

- The Conductor Interaction Method
- The Metaphors
- Learning overhead
- NASA Task Load ratings

The technology evaluation for this stage (use of the eye tracker and gloves) is discussed in 6.5.2.

6.4.2.1 The Conductor Interaction Method

Feedback from the users with regard to the use of the Conductor Interaction Method was mixed. All users said they found it interesting and understood its origins from Human-Human interaction, but a few were hesitant to embrace it as an interaction technique to replace the technique they currently use today. This was particularly the case with the experienced users.

Feedback from the users highlighted that some found the Conductor method to be quite complex, especially when first having to use it. Some of the experienced computer users, in particular, were frustrated with the method mainly due to issues relating to the hardware not being responsive or accurate enough. Other users, as they became more familiar with the method, had a more favourable opinion of it and found the interaction to be an enjoyable experience. The inexperienced users, in particular, were more open to the conductor method and when comparing it to other existing interaction techniques overall they preferred it. The experienced users were split on whether or not they preferred it.

"I didn't have a good time! Didn't get what I wanted, and it took a long time." (Experienced user)

"It was much more fun, much more exciting. There was a sense of doing something novel and seeing how such interaction techniques are capable of working and using them to achieve something; which leads to a belief that they will become more widespread" (Experienced user)

"I can see this becoming the every day user's tool" (Inexperienced user)

"I liked it! Once you get used to it, you can do it really fast!" (Inexperienced user)

"Adjusting Controls was easier with the gloves, it was more natural for me". (Inexperienced user)

"It was really good fun! I really enjoyed it!" (Inexperienced user)

"It would be easier once you have mastered the hands!" (Inexperienced user)

The two-phased interaction used by the conductor method was received particularly well. Users were generally happy and comfortable with the hand-eye coordination required (with regards to selecting via gaze and manipulating via gestures). One user did point out that he had a small attention span and because he is used to using the mouse, he would normally have looked away from a "target" before pointing at it (with respects to selecting a Media Gallery). A number of users commented that at the times when there were no difficulties due to hardware calibration, interaction was smooth and felt natural.

"This type of interaction is quite natural if it works smoothly..." (Inexperienced user)

"When I see something to point at, I would have looked away by the time I move my hand to point" (Experienced user)

The majority of users believed that the Conductor Interaction Method was appropriate as an interaction mechanism for the creation of multi-media presentations. Users believed that the interaction method would be best suited for application domains that relied heavily on manipulation, and suggested alternative applications including browsing, music mixing, gaming and designing. One user also pointed out that the Conductor Interaction Method would be a more suitable means of human-computer interaction for people with disabilities (in particular for those who already heavily use gestures, for example deaf people).

"Using hands to help people with disabilities" (Inexperienced user)

"in a kitchen environment e.g. cook-book browsing" (Inexperienced user)

"In general for any human-computer interaction" (Experienced user)

"Some kind of sorting through objects, or browsing /searching; perhaps a specialist desktop application of some kind." (Experienced user)

6.4.2.2 The Metaphors

Most users understood the use of the Orchestra and the Conductor Metaphors, and found their use appropriate for the nature of the tasks they had to complete. In observing the experienced users, it was evident that due to their experience with the desktop metaphors they had the tendency to refer and compare the Orchestra and Conductor metaphors to the desktop metaphor. The inexperienced users made fewer such comparisons.

The Orchestra Metaphor

The idea of a stage that displayed the Media Galleries and a central stage to interact through was generally well received. Feedback and observation indicated that overall the metaphor was found useful by the majority of the users. Its use assisted them in associating particular media types to particular stage objects and in defining a central area where their interaction and presentation would take form. However, one of the experienced users, who liked the Conductor Interaction Method as a whole, commented that the Orchestra Metaphor "did not work for him" and it was just a different type of "desktop".

Within the Presentation Conductor application, most users found the Media Galleries recognisable and could associate their content with the icons that depicted them. In a couple of cases the Film and Animation galleries were confused, which suggests that perhaps some of the icons should be made more distinguishable, or perhaps some of the galleries could be combined. A few users also commented on the size of the icons and that they would have preferred them to be slightly larger 'targets'.

" I can't seem to be able to hold my focus on it [Media Gallery] for long. Maybe it would have been easier if it was bigger...." (Experienced User)

The use of gaze to give a cue in the way that the conductor would give the cue to a musician and the subsequent gesture interaction was understood by all users. A notable observation was that although this system was designed before the release of the film "Minority Report" (Spielberg, 2002), as the evaluation was carried out after its theatrical release, users who had seen the film would frequently comment the interaction was "just like Tom Cruise in Minority Report". In a way this overshadowed the Conductor metaphor. Despite this most users, especially the inexperienced ones, became very involved with the interaction and *assumed the role* of the Conductor. It was observed that the Conductor metaphor was very helpful in making the user feel part of the system and understand the use of the two-phased interaction.

"Now I really feel like a Maestro!" (Inexperienced user)

Within the Presentation Conductor application the feedback on the gestures used as part of the Conductor metaphor was mixed. Generally users found the majority of gestures natural in the context that they were used, others commented that they could not associate some of the gestures with actions they perform in their every day life. One particular gesture seemed to be difficult for a large proportion of users (***gesture: Both hands vertical flat, palms out***), and from examining the video recordings this seems to be because the users tend to attempt to twist their arms instead of their wrists. A number of users also found some of the gestures quite difficult (and painful) to perform. This highlights the difference in wrist flexibility that people possess.

It was observed that some users would sometimes mirror the gesture that they were performing with one hand, with the other. This happened mainly with the (***gesture: Right hand Point***) gesture, which resulted in the system instead recognising the (***gesture: Hands framing***). It was also observed that while sitting users had the tendency to relax their left hand, when it was not active, in a flat position on the table. This had as an effect that at some point both hands would be recognised as being flat, and interpreted as the (***gesture: Both hands flat***), which depending on the context is used either as a cancel or as a finalisation gesture. These two examples illustrate that either some of the gestures need to be altered, or perhaps mechanisms need to be added to the implementation to prevent this gesture misinterpretation.

Overall, users managed to perform to the majority of gestures, felt comfortable in doing them and were able to associate and recall them in the context that they were used.

6.4.2.3 Learning overhead

In order to use the method, it was first necessary for the users to undergo a period of gesture training as has been described previously. Evaluation feedback indicated that users would have liked a visual representation of their hands during this process, so they could understand what they were doing and when they were getting the gestures wrong. Getting the users to learn the gestures was in some cases a lengthy task, and better feedback could have assisted in the task.

"perhaps you could do with some visual representation of the gestures at this stage [gesture training]" (Experienced User)

"more visual feedback [of gestures] would have been helpful" (Experienced User)

How long it took the users to learn the principles of the method varied. One experienced user interacted with the system perfectly from the outset. With no guidance this particular user was able to find a gallery of his choice, select a media object, edit its volume and then add it to the presentation. In general, however, it was observed that users would initially struggle with the method, but as it became more familiar would find it easier to use. This was highlighted by the fact that the number of questions (concerning the interaction) they asked gradually lessened, and they tended to increasingly become requests for confirmation, such as:

"so now I turn my hand like this [performs left hand twist]" (Experienced User)

"I put my hands flat now!?" (Inexperienced user)

It was also observed that the context of the interaction would often help the users to remember what gesture to use (for example, turning the brightness knob using a (***gesture: left-hand twist***)). This further illustrates how choosing a gesture vocabulary that reflects the desired actions (whether on the interface or from the real world) can have a significant affect on reducing the learning overhead.

*"Oh yes so now I push it like this [performs (**gesture: Right hand vertical flat**)] to move it to the left"* (Inexperienced user)

"Oh yes, the gestures make sense in the context that they are used" (Experienced user)

Overall, users found the learning overhead for the Conductor Interaction Method to be greater when compared to other existing interaction techniques. This was particularly the case for the experienced users though this was likely due to their bias towards the interaction techniques with which they were already familiar. The majority of the inexperienced users, however, believed that the learning required was no different from any of the other (if any) interaction techniques they had used. Although this suggests that currently the Conductor Interaction

Method might not be an easier interaction interface for inexperienced users to use, it does suggest that it is perhaps on a par with existing ones. This is promising, especially when one considers that the implementation of the method is but a prototype and that it and the method could be further refined. As has been discussed, a key refinement would be to provide more visual feedback during the training session to help assist in the interaction learning process.

6.4.2.4 NASA Task Load ratings

Using the NASA Task load Index (TLX), as explained in Appendix D, the following graphs show how the users rated their workload for the two experiments within the first stage. Figure 6.3 shows how the experienced and the inexperienced user groups rated the tasks required for the PowerPoint experiment. Generally all the values are below 10, indicating that very little Mental, Physical and Temporal demand was expected form the user, they did not have to put much effort nor did they get frustrated. Both groups also rated performance relatively good (as shown in Appendix D, the lower the TLX Performance rating the *better* the users regard the performance).

Interestingly, the graph in figure 6.2 shows that the two groups found this experiment to have a relatively similar workload. One possible explanation for this is that although the inexperienced group had no prior experience of PowerPoint (or any similar application), they might have been familiar with the Microsoft Windows environment. As a result they would already be reasonably familiar with the style of interaction. Future evaluation studies could involve a group of *totally* computer illiterate users, although finding such subjects is becoming increasingly difficult.

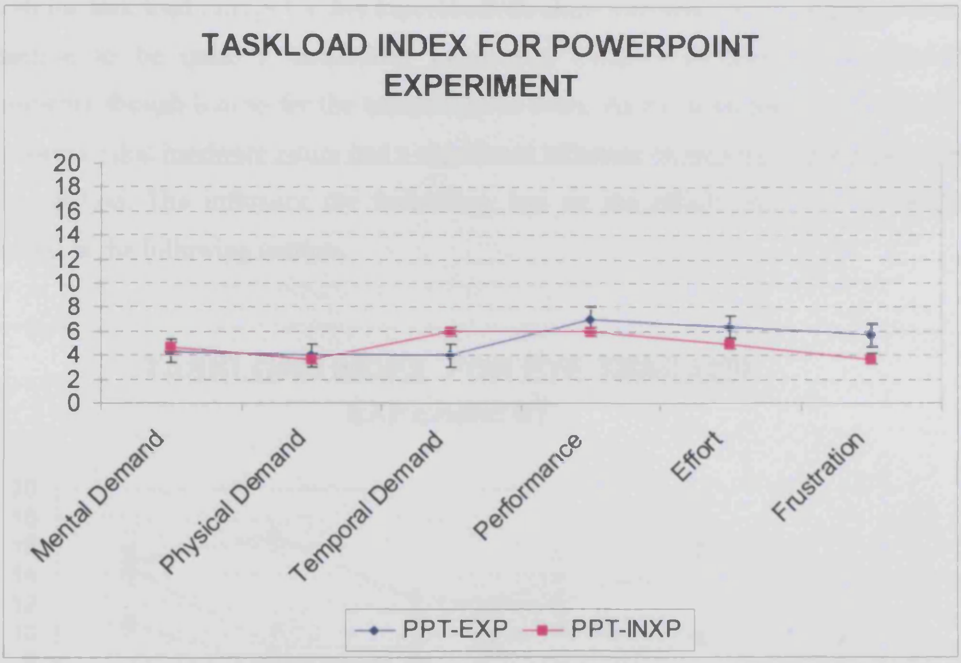


Figure 6. 2 - Taskload for the PowerPoint experiment

Figure 6.3 shows the task load for the eye tracker experiment in which three groups participated (now including the control group). From the graph it is evident that the experienced users encountered the most difficult in adapting to using the Conductor method. In particular the experienced users found the method to be very physically demanding and to require a lot of effort. Obviously, this most likely reflects that they are used to carrying out the same types of operations with an interaction method (i.e. a WIMP interface) and hardware (i.e. the mouse) that they are very familiar with. In particular, the experienced users found the gloves and eye tracker to be very frustrating. The hardware issues are discussed later in this chapter.

Interestingly, although the two inexperienced groups also found this experiment to involve quite a high workload, the two group results were quite different in some respects. The inexperienced control group (who did not take part in the PowerPoint experiment) gave the method an overall workload rating similar to that given by the experience group. The most notable difference between the two being that the control group found that the method made considerably fewer physical demands than the experienced group.

The non-control inexperienced group, however, gave lower workload ratings (averaging around 9). A likely cause for this quite significant difference between the two groups is that the non-control group's ratings for the method were probably influenced by their experiences with using PowerPoint in the first experiment. The control group did not have such bias.

Overall the task load ratings for this experiment do show that users from all groups found the interaction to be quite a demanding experience (when compared to the PowerPoint experiment), though less so for the inexperienced users. As has been previously discussed, it was observed that hardware issues had a significant influence on how users rated the workload of the method. The influence the technology has on the effectiveness of the method is examined in the following section.

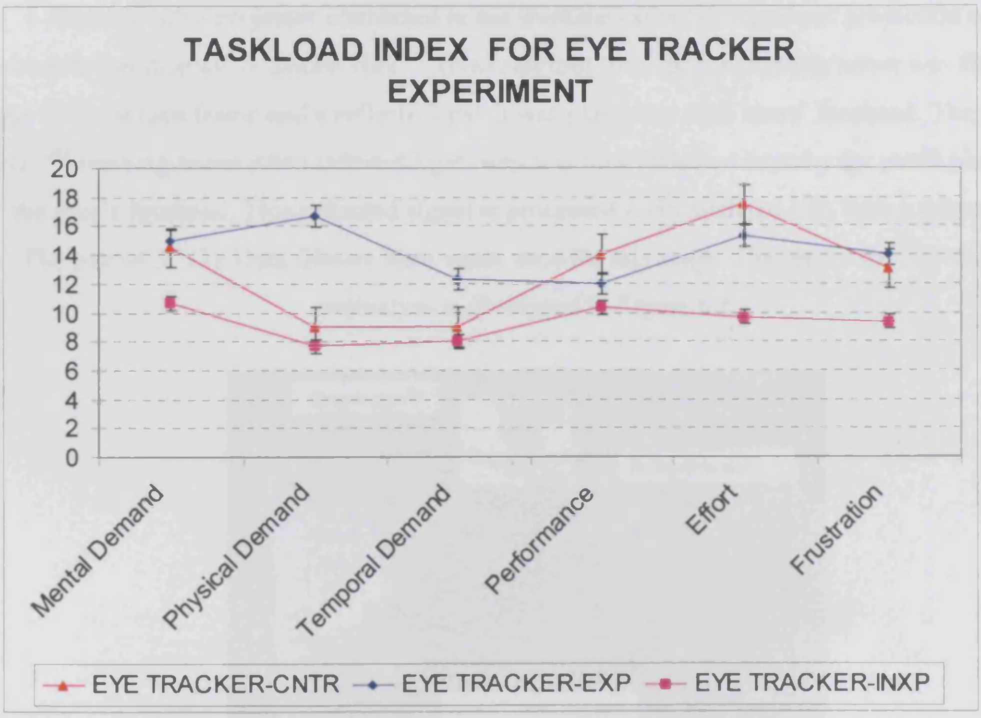


Figure 6. 3 - Taskload for Eye Tracker experiment

6.5 Stage 2: Evaluating the technology

This stage of the evaluation was mainly concerned with obtaining feedback on how different technologies can have an impact on the effectiveness of the Conductor Interaction Method. To some extent the method evaluation continues in this stage, but by now the users have become familiar with the environment and the tasks.

As has been discussed, the Conductor Interaction Method relies on the use of multimodal interaction. For this reason it is important to evaluate the devices and mechanisms used to capture input in order to see whether they can affect the usability and effectiveness of the method. For this evaluation it was decided to assess whether head pointing (instead of gaze), in combination with a large screen would have any effect on the usability of the Conductor Interaction Method. As described in chapter 3, experiments carried out by Stiefelhagen and

Zhu (2002) showed that head orientation could be used to determine the object of a person's gaze, or focus of attention. The findings of this experiment influenced the decision to use this approach to determine the user's focus of attention.

6.5.1 Format of session

For this stage of the evaluation the user was standing in front of a projection screen (1.25m x 1.25m). A video projector connected to the workstation provided a back projection of the workstation display. A *Smart*- Nav^{TM} (NaturalPoint Inc) optical tracking tower was fixed on top of the screen frame and a reflective patch was placed on each users' forehead. The *Smart*-Nav^{TM} tracking tower emits infrared light, which is then reflected back by the patch placed on the user's forehead. This reflected signal is processed and translated into cursor movement. The pair of 5th DT Data Gloves were again used for this stage. The set-up for this stage of evaluation is illustrated in Figure 6.4.

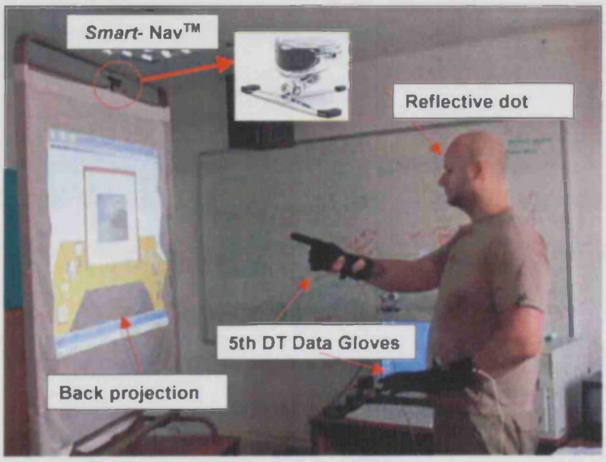


Figure 6. 4 -Stage 2 set-up

The second stage was broken down into two phases in which all groups took part:

- *Technology briefing, calibration and training.* The stage began by briefing the user about the different technology they would be using in this experiment. The use of the *Smart*- Nav^{TM} technology (Natural Point Inc.) was explained and then calibrated to the user (taking into account issues such as user height). The user was given a small training period with the head tracker, so they could see and understand the effects their head movements would have on the mouse pointer. Finally, because for this stage the user was in a different posture to stage one (now standing), it was necessary to re-calibrate the hand gesture software for the new set-up.

- *Presentation Conductor Tasks*. The user was then asked to complete the same set of tasks that they performed in stage one, but this time using the new set-up and technology. As before, the investigator read out the task instructions, to keep the conditions of the experiments consistent and to allow the user to focus on the task at hand. Upon completion of the tasks the user was asked to complete another Task Load Index rating and the remainder of the evaluation questionnaire.

As for stage one, the same evaluation structure was used for all users and was filmed throughout. Again, the users were asked questions and encouraged to voice their comments.

6.5.2 Discussion of sessions

This section provides a discussion of the experiences in using the different technologies within the evaluation and has been analysed in the following categories:

- Experiences with the Eye Tracker and Gloves
- Experiences with the Large Screen and Gloves
- NASA Task Load ratings

The experiences relating to the combination of eye tracker and gloves stem from stage 1 of the evaluation. The experiences relating to the combination of the head tracker and gloves come from stage 2.

6.5.2.1 Experiences with the Eye Tracker and Gloves

As has been detailed in section 6.4, in stage 1 of the evaluation users had to utilise an eye tracker and data gloves as part of the method. Most users (both inexperienced and experienced) had not used an eye tracker before and none had used data gloves. Despite this, all users appeared enthusiastic about making use of these two technologies as a mechanism for interaction. Although in general the hardware worked well and succeeded in achieving what was desired from it, there were a number of problems that were significant enough to affect the users experience of the method. These problems were consistent across all groups and were mainly due to the limitations of the specific hardware rather than the way that the users were interacting.

The data gloves did not present many problems. However, in one case they failed to work during a session, but upon inspection it was discovered that this was due to a loose optical fibre and was quickly remedied. All users said that they hardly felt any weight on their hands and wrists from the gloves, and many of them commented that after a few minutes they forgot that they were actually wearing them. The size of the gloves was not a particular problem, except for a few cases where the users had short fingers and as a result the gestures were not always recognised. As a consequence of this, it meant that users tended to pay more attention in forming certain gestures to reduce the recognition problem. Additionally, it was also observed that when performing a point gesture users found it much easier to use their thumb to hold the other fingers in a fist than to have the thumb open and the remaining fingers free to move.

The eye tracker, on the other hand, was the most problematic piece of hardware used in the evaluation, with almost all users experiencing difficulties using it. Particularly problematic was the eye tracker calibration that was required for each user. Most users needed a number of attempts in order to achieve an acceptable calibration, and even when calibration was achieved it would frequently be lost during the experiments. Users who wore glasses found it particularly hard to calibrate the eye tracker, and in two cases calibration could not be achieved at all. However, other glass wearing users were able to achieve a calibration, suggesting that the type of lens used within the glasses was an important factor.

> *"The eye tracker didn't work for me. The gloves failed to find certain positions..."*
> (Experienced user)

During the experiment the eye tracker calibration was lost for a significant number of users. The most common reasons for this were observed to be:

1. Some users were fidgety and kept readjusting their seating position, inevitably moving form the position that the eye tracker was originally calibrated for.

2. As the users performed the gestures their body would move and this would again result in them moving out of position.

3. A number of users did not change the focus of what they were looking at by moving their eyes but instead by moving their head. This was the most common reason for calibration loss.

When calibration was lost the users had to go through another period of recalibration. Eventually most users overcame their problems but there were a few that had problems through out the experiment and had to recalibrate the eye tracker up to five times.

In general this combination of technology was problematic. This was mainly due to the eye tracker, which was very restrictive as far as user movement is concerned. On the other hand, the performance of gestures often resulted in body and head movement of the user and consequently the eye tracker lost its calibration. The fact that the eye tracker required repeated calibration made the whole process intrusive and frustrating for the user. Future work could involve performing additional evaluations using alternative eye trackers that have a pan and tilt function, such as the ASL504, possibly combined with a magnetic head tracker, to see whether this situation could be improved. The data glove technology was far less problematic, and generally integrated well with the method. The gloves were fairly robust and lightweight, causing no particular problems throughout the experiment.

6.5.2.2 Experiences with the Large Screen and Gloves

As with the eye tracker the majority of users had not used an optical tracking tower before, however of the two, it was clearly the preferred set up. Removing the tracking technology from the user's person not only reduced the likelihood of technical difficulties, but also allowed the user much greater freedom of movement. Of all possible factors, it was probably this freedom of movement that contributed most to the optical tracking tower's success as an alternative to the eye tracker.

"In the beginning I preferred the eye tracker, but when I saw I could move easier I liked this one better." (Experienced User)

The greater freedom also helped many users to perform the hand gestures. In particular, those that struggled to perform gestures in the sitting position were able to perform them when standing up. The fact that they did not have the desk in front of them also affected the way in which the gestures were performed, a notable example being a reduction in the accidental occurrence of the hands flat gesture (due to the user's hands being rested on the desk).

"Gestures are much easier standing up" (Experienced User)

The use of the optical tracking tower also allowed the user to move their eyes independently of their head, allowing them to use their peripheral vision. On the negative side this meant that if a user looked at an object without moving their head this would not be detected by the optical tracking tower. However, no users reported this as being a problem, presumably because a large screen was used.

"You can use peripheral vision now... this is much better" (Experienced User)

The optical tracking tower itself benefited from not requiring any user calibration (and thus no recalibration during the experiments), which was a major issue with the eye tracker. It took most users only a few minutes to get used to moving the pointer around the screen with their head, in contrast to the larger calibration/training period with the eye tracker. Only one user, who was particularly tall, had problems with the large screen set-up, but even he preferred this set-up to the eye tracker.

The use of the large screen was also found to be more favourable. The sizes of the displayed Media Gallery objects were larger and consequently users found them easier to select. One negative aspect of using back projection was that at some positions the light of the projector would dazzle the users, resulting in a blind spot on the screen. One possible solution to this would be to use different large screen technologies such as a plasma screen.

"This [big screen set up] is far better" (Experienced User)

"Blind spot on the projection when looking directly at projector" (Experienced User)

"Having the light in my face is a bit annoying" (Experienced User)

One other issue that was raised was that of having to stand up all the time whilst performing the interaction. One user complained that his legs became increasingly tired during the evaluation. A simple solution to this would be to allow the user to sit on a stool or something similar.

Overall, the majority of users found the interaction smoother with this set up, were far more confident when doing the tasks, and in general preferred it as an interaction mechanism. This set-up also benefited from producing far fewer technical issues.

6.5.2.3 NASA Task Load ratings

Figure 6.5 shows the task load ratings for the large screen experiment in which all three groups participated. From the graph, it is evident that overall all groups found the head tracker set up to require less workload to use than the eye tracker (c.f. figure 6.3). As before, both the experienced and control groups reported the highest workload ratings, and again the average ratings of the two groups were similar. Interestingly, in this stage the control group found the head tracker set up to require considerable physical effort but performed very well, almost a complete reversal of how they rated the eye tracker set up.

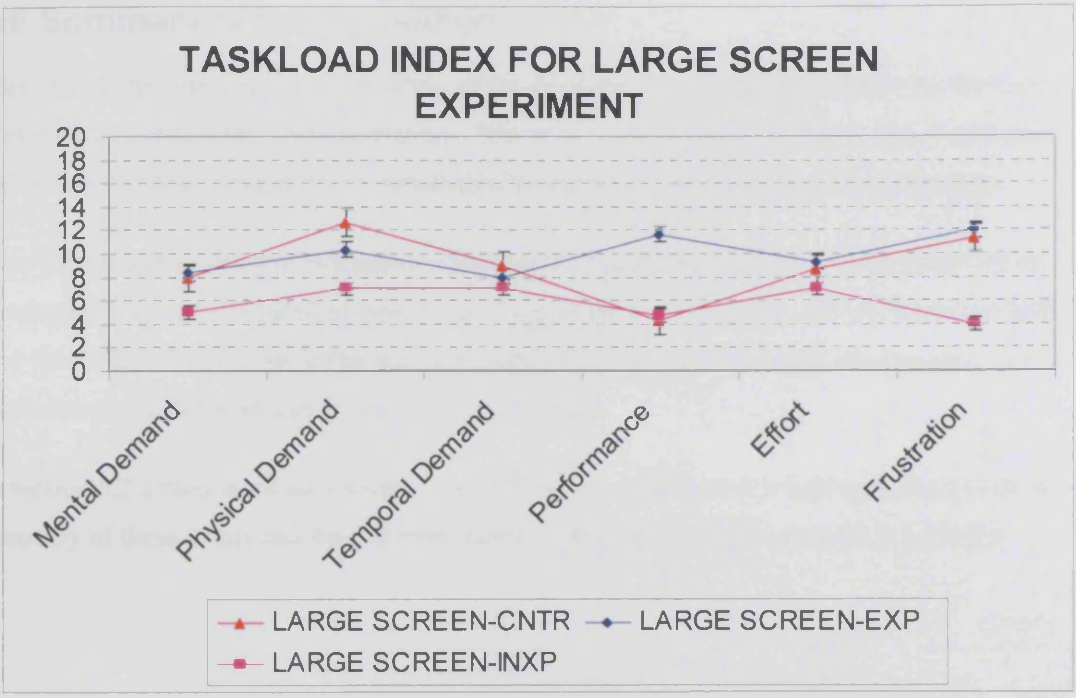


Figure 6. 5 – Taskload for Large Screen experiment

In general, the average workload ratings provided by these two groups whilst using the large screen set up was noticeably lower than required for the eye tracker set up; averaging around 9 - 10 in comparison to 13 – 14 for the eye tracker. This illustrates that the choice of technology can have a significant impact on the perceived workload of the Conductor Interaction Method.

The inexperienced group also found the large screen set up to possess a lower workload overall and again provided a set of fairly steady ratings. As with the eye tracker set up, there is quite a significant difference between the control group and the inexperienced group. This again possibly reflects the initial PowerPoint experiment having an influence on the inexperienced group's later ratings. In general, the average workload ratings provided for the large screen set up were around 6 in comparison to around 9 for the eye tracker. More positive is that this is a very similar value to how the inexperienced group rated PowerPoint (around 5). This is very promising considering the technology and implementation of the method can still be further refined.

Overall the task load ratings for this experiment showed that all users found the head tracker set up to possess a lower workload than the eye tracker set up. As has been previously discussed, a significant reason for this is probably the fewer technical difficulties and constraints that were experienced. However, that the users were at this stage more familiar with the method and the gesture vocabulary also probably contributes to these results to some extent.

6.6 Summary of the evaluation

This chapter has presented an evaluation of the developed Conductor Interaction Method and Presentation Conductor implementation. The evaluation aimed to assess the Conductor Interaction Method through the Presentation Conductor, its usefulness and its application.

A formative and qualitative evaluation approach was used, due to the prototype nature of the developed tool, and consisted of two stages. To provide a quantitative view of the model and tool the NASA Task Load Index was also used. This allowed workload comparisons to be made across the different user groups and experiments.

In section 6.2 a number of issues were identified that the evaluation would seek to address. A summary of these issues and the relevant feedback that was obtained is shown in table 6.3.

Issues	Feedback
If users find it easier to interact with computers using the Conductor Interaction Method, than with existing interaction techniques	Users found the Conductor Interaction Method to be complex, especially when first using it. Despite this, the two-phased interaction used by the conductor method was received particularly well. The experienced users were less favourable about the method. **(cf. section 6.4.2.1)**
If the metaphors that comprise the method are useful and beneficial	Both metaphors were found useful by the majority of users, although some experienced users misinterpreted the Orchestra metaphor. Users found the Conductor metaphor to be very engaging and helped in their understanding of the Conductor Interaction Method. Both, however, could still be further refined **(cf. section 6.4.2.2)**
If the learning overhead for the method is lower than with existing interaction techniques	Experienced users found the learning overhead for the Conductor Interaction Method to be greater when compared to existing interaction techniques. Inexperienced users found the learning overhead to be no different from any other interaction technique they had used. **(cf. section 6.4.2.3)**
If users would prefer to use such an approach over existing ones	Experienced users were split on whether they preferred the Conductor Interaction Method to existing techniques, while the inexperienced users in general preferred it. **(cf. section 6.4.2.1)**
If the technology used can influence the effectiveness of the Conductor Interaction Method, in particular gaze versus head pointing, and small screen versus large screen	It was observed that the effectiveness of the Conductor Interaction Method could be significantly influenced by the technology used. This was particularly the case with the eye tracker, which repeatedly required calibration. The large screen set-up was generally more effective, with almost no problems **(cf. sections 6.5.2.1 and 6.5.2.2)**
If the Conductor Interaction Method is appropriate for this type of application, and whether users would prefer to use such an interaction approach for such an application	Users found the Conductor Interaction Method was appropriate for this type of application. Experienced users were split on whether they would prefer to use such an interaction method, while the inexperienced users preferred it. **(cf. section 6.4.2.1)**

Table 6. 3 - Summary of evaluation issues and feedback

Feedback on the method was mixed. As expected, it was less well received by the experienced users, most likely due to their familiarity with, and bias towards, alternative interaction methods (i.e. WIMP). However, the inexperienced users found it an enjoyable experience and were open to the idea of using it rather than other interacting techniques. This was more positive as one of the initial objectives for the method was that it should be most beneficial to users with little or no computing experience. Although the feedback was not overwhelmingly positive, it is promising that an alternative interaction method, while at the prototype stage, was generally well received.

Assessment of the different technologies indicated that the effectiveness of the method could be significantly affected by technology that is used. In particular, technology that is restrictive or prone to technical problems (for example, recalibration) can be a major impediment to smooth interaction. It would be worthwhile to evaluate other available technologies with the Conductor Interaction Method to further assess the impact technologies can have on the method.

The task workload assessment that was carried out also illustrated the effect the technology could have on the method, as well as highlighting the perceived differences between experienced and inexperienced groups. Figure 6.6 summarises the task workload for all the experiments. The graph also helps to illustrate how the ratings provided by the inexperienced group for the PowerPoint and large screen experiments were on a similar scale.

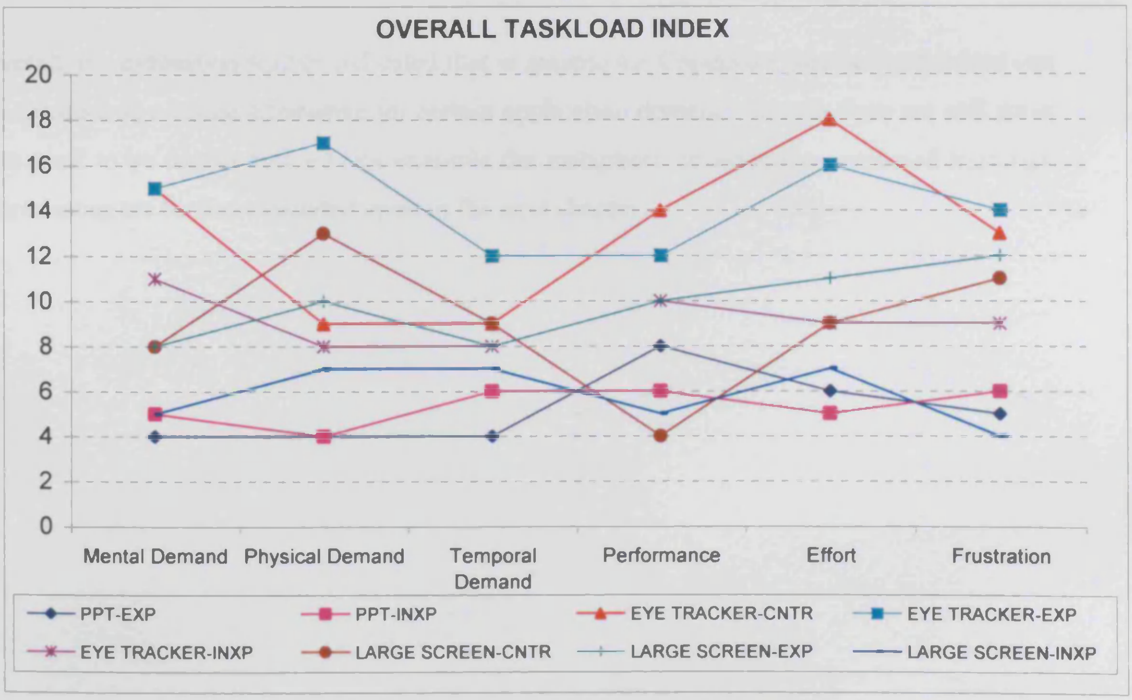


Figure 6.6 - Taskload for all experiments by all groups

The Presentation Conductor itself was generally well received and deemed as being an appropriate application for the method (again, the experienced users were less disposed towards it). Other possible application domains suggested by the users included browsing, music mixing, gaming and designing. Interestingly, these domains share the fact that in a non-computing environment they would all typically involve the use of hand-based manipulations. This highlights the fact that the Conductor Interaction Method is probably most suitable for applications where a hand-based interaction approach would be central, and possibly where an equivalent or similar interaction is already carried out in a non-computing environment.

One issue that was raised was the amount of training required in order to use the application (and method). Better training support was suggested as a possible solution. It was observed, however, that once the users had begun using the application, the interaction would become easier and the users would more readily recall the appropriate gestures based on the context.

It would have also been interesting to see what effect the order of the tasks within the evaluation, could have on the results. This would illustrate whether the user's experience and preference of technology is influenced by the order that the experiments are conducted and whether they can adapt to using the eye tracker after using the optical tracking tower. Unfortunately, as it was difficult to recruit larger numbers of inexperienced users, this could not be investigated. Three additional groups of each type, would be needed to perform to the evaluation in reverse order, i.e. large screen experiment first and eye tracker last. Future evaluations could place more of an emphasis on investigating this, however.

Overall, the evaluation studies indicated that in general the Conductor Interaction Method can be consider as a viable alternative for certain application domains, though there are still areas that need to be further refined (for example the metaphors, technologies used and training). These areas are further expanded upon in the next chapter.

Chapter 7 – Conclusions

7.1 Summary of thesis

This thesis has proposed a novel interaction method, the Conductor Interaction Method. It has been argued that existing interaction techniques have three notable failings. They are typically unnatural and require the user to learn a new interaction method, they are often inconsistent in their semantic language, and they can restrict creativity and be over simplistic. The proposed Conductor Interaction Method seeks to provide an alternative interaction technique that can tackle some of these failings. To achieve this it makes use of a two-phased multi-modal interaction mechanism, gaze for selection and gestures for manipulation, as well as incorporating this in a metaphor based environment

Chapter 2 examined a number of interaction techniques and methods used for interacting with a computer. Eight of the more common techniques were briefly reviewed, and their failings discussed. It was argued that the majority of these failings could be overcome by drawing upon techniques and experiences from Human-Human interaction. Two such interaction techniques, gaze and gestures, were examined, as well as the mechanisms that have been developed to support their use within Human-Computer Interaction. It was argued that these modalities were appropriate for the novel interaction method presented in this thesis.

In chapter 3 nine key systems that use gaze, gesture or their combination as a means of interacting were examined. The analysis of these systems highlighted a number of issues that were considered in the design of the Conductor Interaction Method and later in the development of the Presentation Conductor architecture. More specifically, the Midas touch problem was addressed and a two-phased interaction style was adopted.

In chapter 4, the Conductor Interaction Method, the interaction approach proposed by this thesis was presented. The Orchestra and the Conductor metaphors that play an important role in the method were described. It was also discussed how two-phased interaction, a more natural interface, and the reduction of learning overhead have been provided for in the method. The architecture for the Presentation Conductor, a system that realises the interaction method, was also presented. The Presentation Conductor application allows users to create multi-media presentations via the use of the Conductor Interaction Method. Scenarios were provided to illustrate the system in use.

Chapter 5 discussed the implementation of the Presentation Conductor prototype and how the Conductor Interaction Method was implemented within it. A detailed description of the system was provided, with examples used to illustrate its functionality.

Chapter 6 presented the evaluation of the Conductor Interaction Method that was carried out in order to assess the usefulness of the method as an alternative interaction approach. The evaluation also seeked to examine the effect the choice of technologies could have on the applicability of the method. A formative and qualitative evaluation was performed, that involved users with differing computer experience using the Presentation Conductor to create a presentation. The evaluation showed that the Conductor Interaction Method can lead to a viable alternative interaction technique, especially for inexperienced computer users. However, there are still areas, in particular within the implementation, that could be further refined.

This chapter seeks to conclude on the work presented in this thesis by first returning to the objectives that were outlined in chapter 1 and discussing how these have been satisfied. The chapter then moves on to highlight possible areas of future work.

7.2 Thesis Objectives revisited

In chapter 1, three objectives were set out for the work proposed in this thesis. These objectives will now be revisited, and it will be discussed how they have been met.

7.2.1 Objective 1

"To develop a method for interacting with computers that utilises modalities that are typically used in Human-Human communication, such as gaze and gesture. The method would support the use of these novel interaction techniques with a specially tailored interface based on metaphors that users can relate to. Key requirements for the method would be for it to possess a low learning overhead and be easy to use."

This thesis has proposed the Conductor Interaction Method as an alternative and novel interaction technique. This method utilises both gaze and gesture modalities as a means of interacting with the computer, and supports this with the use of specifically created Conductor and Orchestra metaphors.

The Orchestra Metaphor provides the user with an environment to interact in, by presenting graphically the available resources that can be manipulated. The metaphor has it origins in the theatre set-up; where the stage represents the main interaction and presentation area, and the musicians below the stage represent the resources that are available to the user. A number of benefits arise from using the Orchestra metaphor: the visual representation of the resources used caters well for gaze-based interaction, and the familiarity of the user with the stage set-up gives a sense of expectation and helps in orienting him/her with the interface.

The Conductor metaphor is interaction orientated and is used in conjunction with the Orchestra metaphor. In a similar manner as a conductor interacts with the members of an orchestra, the user can interact with the resources. Interaction is first established via gaze (eye contact) before using gestures to manipulate the resource (musician). The Conductor metaphor is specifically tailored for gaze and gesture based interaction and so is more suitable than other existing HCI metaphors, and when used in conjunction with the Orchestra metaphor results in an interface that is easier to use and understand, especially for novice users.

With the above two metaphors, the Conductor Interaction Method is able to provide users with an interaction technique that possesses a low learning overhead and is easy to use. Gaze and gestures are interaction mechanisms that are typically used by humans in their every day human-human interaction, and by combining the two the user is able to perform both simple and complex interactions. A low learning overhead for the method is also encouraged through support for gestures that the user can easily associate with activities they perform in their everyday life.

7.2.2 Objective 2

"To demonstrate the feasibility of the method by designing and implementing a prototypical system, the Presentation Conductor, that would utilise it as the main method of interaction."

The Conductor Interaction Method is represented in the Presentation Conductor system. With this system the user can create and display multi-media presentations. The architecture of the Presentation conductor, as presented in chapter 4, consists of three major components: the User, the Interfaces and the Application. The User interacts with the system to create a presentation and embodies the Conductor metaphor. The Interfaces (the gesture, the gaze and the audio-visual interface), enable the user to interact with the application. The Application enables the user to create a multi-media presentation by interacting through the gaze and gesture Interfaces. The Conductor and Orchestra metaphors are used to support the user to create a presentation. The Orchestra metaphor is visualised and presented to the user via a computer display using the audio-visual interface. The stored media objects that are used to compose the presentations, and previously created presentations are also managed by the Application.

The developed prototype system utilises the gaze and gesture modalities to perform sophisticated activities. Both the Orchestra and the Conductor metaphor have been implemented. A significant number of gestures can be implemented in the realisation of the conductor metaphor, while the realisation of the Orchestra metaphor provides the environment for the multi-media presentations to be created. The implementation supports two-phased interaction, where first gaze is used for selection and then gestures are used in order to manipulate and control. The implementation attempts to reduce learning overhead by using gestures that are familiar to those used to perform the task in question in analogous real world situations.

7.2.3 Objective 3

"To evaluate the Conductor Interaction Method through the Presentation Conductor. The evaluation should compare the method against existing interaction approaches to see whether it can be a beneficial alternative. The evaluation studies should involve a range of users with differing computer experience in order to provide a broad study."

Chapter 6 provides an evaluation plan and analysis of the evaluation carried out.

A formative and qualitative evaluation approach was carried out in two stages. This approach was taken due to the prototype nature of the developed tool and because HCI interactions are particularly subjective in nature. To provide a quantitative view of the model and tool the NASA Task Load Index was also used. This allowed workload comparisons to be made across the different user groups and experiments.

The first stage of the evaluation aimed at obtaining feedback on the Conductor Interaction Method and more specifically on making a comparison with other interaction techniques. For the evaluation a comparison was made with Microsoft PowerPoint. The second stage was focused on obtaining feedback on how different technologies could have an impact on the effectiveness of the Conductor Interaction Method.

Feedback from the evaluation with regards to the method was mixed. Although it was not overwhelmingly positive, there were indications that it could be considered as an alternative interaction method if further refined (the implementation in particular). With regard to the technologies used, the feedback indicated that the effectiveness of the method could be significantly affected by the technology used. In particular technology that is intrusive and restrictive can have a significant negative impact on users' ability to perform the interactions.

The evaluation studies that were performed satisfy the goals that were set in this objective. The NASA Task Load Index provided a form of quantitative analysis of the qualitative results, by measuring the workload that users perceived themselves to be under. This was done as an alternative to attempting to do a full quantitative evaluation that, as discussed, was deemed inappropriate for a system of this nature.

The order effects were not investigated in this evaluation, but it would have been interesting to see how users would find the use of the eye tracker after having the experience of the large screen, as they would already be familiar with the Conductor Interaction Method and have learnt the gesture vocabulary. Additionally it would be interesting to see what other effects the different order would have

7.3 Summary of Contributions

The work presented in this thesis represents a novel approach to Human-Computer Interaction. The specific contributions of the work can be summarised as follows:

- *The Conductor Interaction Method.* This thesis presents a novel interaction method, which aims to provide users with a more natural way of interacting with a computer, and seeks to overcome some of the failings exhibited by existing human-computer interaction techniques.

  The presented method, the Conductor Interaction Method, specifies the adoption in HCI of human-human multimodal interaction mechanisms, namely gaze and gesture. The method also provides the foundation for a novel interaction environment that is based on the Orchestra and Conductor metaphors.

- *Orchestra and Conductor metaphors.* This thesis presents two novel metaphors that have been developed in order to provide the user with an environment to interact in and a technique with which to perform the interaction.

  The Orchestra metaphor provides the user with a 'stage' based environment in which they can perform their interactions by using a combination of gestures and gaze. The Orchestra metaphor graphically presents to the user the resources that are available for them to manipulate, as well as providing an area in which to perform these manipulations. The familiarity of the stage set-up gives the user the sense of expectation and helps in orienting them with the interface.

  The Conductor metaphor is an interaction metaphor that is used in conjunction with the stage-based environment with a view to providing the user with an interface that is easier to use and understand, especially for inexperienced computer users. As the name suggests the user interacts with the resources depicted within the Orchestra metaphor in the same way that a conductor would interact with the musicians of an orchestra. The Conductor metaphor has been specifically designed for gaze and gesture based interaction methods.

- *An architecture and implementation for a prototypical system to demonstrate the Conductor Interaction Method.* This thesis presents an architecture for a prototypical system that demonstrates the Conductor Interaction Method. An implementation has also been developed that realised this architecture. The developed system, the Presentation Conductor, allows users to construct and display multi-media presentations by making use of techniques specified in the Conductor method. This system also forms the basis of the evaluation that has been carried out that aims to assess the applicability of the Conductor Interaction Method.

- *Experience of using the method.* This thesis has presented an evaluation of the Conductor Interaction Method via use of the Presentation Conductor system. The evaluation primarily aimed to assess the usefulness of the Conductor Interaction Method and to obtain feedback on its application. It also provided an initial assessment of the impact different technologies could have on the method's effectiveness. Experiments were carried out over two stages in which users interacted with systems based on the Conductor Interaction Method but with different hardware set-ups. The user's experiences and feedback were documented and from the analysis areas of possible further development were identified. These are discussed in the following section.

7.4 Further Development

The feedback from the evaluation performed on the Conductor Interaction Method through the Presentation Conductor system highlighted several issues that should be addressed in future work. These issues are examined in this section.

7.4.1 Improved method training support

From the evaluation study one of the most frequently suggested enhancements was better support for gesture training. Users suggested that they would have preferred to have a visual representation of their hand movement, and feedback on the correctness, or otherwise, of the gesture.

A solution to this would be to change the interface to provide a virtual model of the user's hands, and then use visual or audio feedback to confirm when the user has correctly formed the gesture. This would help the user to make the appropriate small adjustments needed sometimes to refine the original gesture. Additionally, there could be an option for the visual feedback to be displayed (for example in top right corner of the screen) throughout the session, if the user requires. This would build up the user's confidence in performing the gestures. A further option would be to provide a small gesture-training program, akin to the tutorials that accompany some complex computer games.

7.4.2 Refining the metaphors

The use of the Orchestra and Conductor metaphors in the Conductor Interaction Method was shown to be quite effective within the evaluation. The feedback, however, illustrated areas in which the metaphors could be further refined to enhance their effectiveness.

With the Orchestra metaphor most users had no problem in associating the icons used to represent the Media galleries with their content, but there were a number of cases where some of the galleries were confused (for example the Film and Animation galleries). This suggests that perhaps some of the icons could be made more distinguishable, or that perhaps in some cases the galleries could be combined. Furthermore, as discussed in chapter 6, users also commented on the small size of the icons, and another refinement would be to make them larger 'targets'. In general, the categorisation of media into galleries is an area that could be further investigated. In particular there is the trade off between the number of galleries versus the number of media files, and how this can affect usability and information overload.

With regards to the Conductor metaphor, feedback showed that on the whole most users were comfortable with the majority of gestures. There were, however, a small number of gestures that some users could not associate with actions that they would perform in their everyday life (for example, some users reported that they never used the ***Both hands vertical flat, palms out*** gesture). Further experiments could be carried out to help determine a vocabulary that is more concise and easier to relate to. One possibility would be to allow the users themselves to define the gestures they wish to use for the different operations. This customisation can be important given that, as shown in the evaluation feedback, not all users are comfortable with all types of gesture.

7.4.3 Experimenting with different technologies

The implementation of the Conductor Interaction Method presented in this thesis, has only made use of a small number of available gesture and gaze based technologies. Further experiments could be carried out using different technologies to see if some of the hardware issues (for example, the eye tracker calibration) can be reduced or removed entirely, as well as to further assess the impact different technologies can have on the applicability of the Conductor Interaction Method.

The eye tracker (LC-Technologies EyeGaze) used in the implementation presented in this thesis was particularly intolerant of user movement, and users would frequently need to recalibrate it. Alternatives exist, including the ASL 504 Pan/Tilt (Applied Science Laboratories), which may be better suited to be used with the Conductor Interaction Method as it can make small adjustments to compensate for the inevitable user head movements that will occur. Additionally, an Ascension Magnetic Head Tracker (Ascension Technology Corporation) could be used in combination with this eye tracker. This would allow for a greater freedom of movement, as the eye tracker would adjust based on the position given by the head tracker.

The use of computer vision for the gesture recognition is one area that could also be explored. This would provide the additional ability to determine the hand position, which could prove beneficial to the gesture refinement. Furthermore, other problems such as the glove fit could also be potentially resolved. The downside of such an approach, however, is that the user may be restricted in their movements given that the cameras used to recognise the gestures will typically be covering specific areas.

Other experiments could also involve position trackers on the hands to determine hand positioning.

7.4.4 Investigating different application domains

User feedback suggested other possible application domains for the method including browsing, music mixing, gaming and designing. All these applications heavily rely on manipulation (typically replicating activities that would be performed by the hands in a non-computing environment), where the use of two-phased interaction like that supported by the Conductor Interaction Method could be appropriate. Further applications could be developed to assess the use of the method within such domains. In turn, acquired feedback could further refine the method and implementations.

Another suggestion arising from the evaluation was to investigate the use of the Conductor Interaction Method as a means of human-computer interaction for people with disabilities. A large proportion of people with disabilities, in particular deaf people, use gestures to interact in their every day life. As well as relying on signing and visual contact, deaf people also use eye contact to establish the communication channel. This combination of gaze and gestures mirrors the two-phase interaction used within the Conductor Interaction Method suggesting that the technique could be an HCI method that they could easily adapt to and utilise.

7.4.5 Further experiments with inexperienced users

The inexperienced users that took part in the evaluation nevertheless had some prior experience with computers. Further experiments could be carried out with *truly* inexperienced computer users, although finding such subjects is becoming increasingly difficult. Possible subjects for these experiments, however, could be young children or the elderly, as these two age groups are less likely to have experience with computers.

7.5 Final comments

This thesis has presented an argument for the consideration of an alternative human-computer interaction method, the Conductor Interaction Method, which provides a more natural interface than that possessed by existing interaction methods. This is achieved by using, in an HCI environment, human-human multimodal interaction mechanisms, notably, gaze and gestures. An architecture for a system that supports this style of interaction, the Presentation Conductor, was developed and a prototype of it, implemented. Finally the method and the use of technology to support it, were evaluated.

The evaluation showed that the Conductor Interaction Method can provide a viable alternative interaction technique, in particular for users with little or no prior computer experience, and within certain application domains, which in their non-computing form typically involve hand manipulation to a great extent

References

Adobe Systems Incorporated, San Jose, California, USA

Angus M., Bettio F., Giachetti A., Gobbetti E., Zanetti G., Zorcolo A. (2002) Real-Time Haptic and Visual Simulation of Bone Dissection, IEEE Virtual Reality Conference, IEEE Computer Society Press, February 2002

Apple Computer, Inc., Cupertino California, USA, http://www.apple.com/

Arfib D., Kessous L. (2002) Gestural Control of Sound Synthesis and Processing Algorithms, In Wachsmuth I and Sowa T., (eds) Gestrure and Sign Language in Human-Computer Interaction, International Gesture Workshop, GW 2001, Springer Lecture Notes in Computer Science, Vol. 2298, Springer-Verlag, Heidelberg, Germany

Argyle M., (1996) Bodily Communication, 2nd Edition, Routledge, London

Armstrong D., Stokoe W., Wilcox S., (1995) Gesture and the Nature of Language, Cambridge University Press, Cambridge

Ask Jeeves – http:// www.ask.com

Ascension Technology Corporation, Burlington, USA , http://www.ascention-tach.com

ASL- Applied Science Laboratories, Bedford, Massachusetts, USA, http://www.a-s-l.com/

Association for Computing Machinery, Special Interest Group on Computer-Human Interaction (ACM SIGCHI)

Bauer B., Kraiss K. F., (2002) Towards an Automatic Sign Language Recognition System Using Subunits, In Wachsmuth I and Sowa T., (eds) Gestrure and Sign Language in Human-Computer Interaction, International Gesture Workshop, GW 2001, Springer Lecture Notes in Computer Science, Vol. 2298, Springer-Verlag, Heidelberg, Germany

Bauml B. J., Bauml F.H., (1997) Dictionary of Worldwide Gestures, Scarecrow Press, Inc., Maryland, USA

Beacker R.M., Grudin J., Buxton W.A.S., Greenberg S., (1995) Readings in Human-Computer Interaction: Toward the Year 2000, Second Edition, Morgan Kaufmann Publishers, San Francisco, California, USA

Benford S., Snowdon D., Greenhalgh C., Ingram R., Knox I., Brown C., (1995) VR-VIBE: A Virtual Environment for Co-operative Information Retrieval, *Eurographics'95 Conference proceedings*, Maastricht, The Netherlands

Beringer N., (2001) Evoking Gestures in SmartKom – Design of the Graphcal User Interface, In Wachsmuth I and Sowa T., (eds) Gestrure and Sign Language in Human-Computer Interaction , International Gesture Workshop, GW 2001, Springer Lecture Notes in Computer Science, Vol. 2298, Springer-Verlag ,Heidelberg, Germany

Bolt R., (1980) "Put-that-there": Voice and gesture at the graphics interface, Proceedings of the 7th annual conference on Computer graphics and interactive techniques, Seattle, Washington, USA

Bolt R. and Harranz E.(1992) Two-Handed Gesture in Multi-Modal Natural Dialogue, UIST'92, November 15-18 1992

Borchers O. (1997) WorldBeat: Designing a Baton-Based Interface for an interactive music Exhibit, CHI97, Jan 1997

British Airways – www.ba.com

Carroll, J.M., Mack R.L. and Kellogg W. (1998) "Interface Metaphors and User Interface Design" in Helander (ed.) Handbook of Human-Computer Interaction, Elsevier Science Publishers B.V., North Holland.

Cooper A. (1995), originally published in Visual Basic Programmer's Journal, available online http://www.cooper.com/articles/art_myth_of_metaphor.htm

COVEN Consortium http://coven.lancs.ac.uk/

CyberGlove Reference Manual (1998), Virtual technologies Inc

Discreet, Discreet /Autodesk, Montreal, Quebec, Canada, http://www.discreet.com/index-nf.html

Dix A. J., Finlay J. E., Abowd G. D., Beale R., (1998) Human-Computer Interaction, 2nd Edition, Prentice Hall Europe, Hertfordshire, UK

Dix A. J., Finlay J. E., Abowd G. D., Beale R., (2004) Human-Computer Interaction, 3rd Edition, Prentice Hall Europe, Hertfordshire, UK

Donald M. (1991) Origins of the Modern Mind: Three Stages in the Evolution of Culture and Cognition, Harard University Press, Cambridge, Mass.

Eco, U. (1984) "Semiotics and the Philosophy of Language", Bloomington, U.S.A.: Indiana University Press

Electronic Arts, Redwood City, California, USA

Efron D. (1972) Gesture, Race and Culture, Mouton, The Hague

Eye-movement equipment database, Institute of Behavioral Sciences, University of Derby, Derby UK

Farid, M., M., and Murtagh, F., (2002) Eye-movements and Voice as Interface Modalities to Computer Systems, SPIE Regional meeting on Optoelectronics, Photonics and Imaging (Opto Ireland), Galway, Ireland, 5-6 September 2002.

Farid, M., M., and Murtagh, F., Starck J. L., (2002) Computer Display Control and Interaction Using Eye-Gaze, Journal of the Society for Information Display, 2002

Fehr, B.J and Exline, R.V. (1987) Social Visual Interaction: A conceptual and Literature Review. In Siegman A.W. and Fieldstein S. (eds) Nonverbal Behavior and Communication, 2nd edition, Halsted, New York

Gentner D., (1983) Structure mapping: A theoretical framework for analogy. Cognitive Science, Volume 7

Gips J., Olivieri P. (1996) EagleEyes: An eye Control System for Persons with Disabilities, The Eleventh International Conference on Technology and persons with Disabilitied, Los Angeles, California, USA, March 1996

Glenstrup, A.J., Engell Nielsen, T., (1995) Eye Controlled Media: Present and Future State, Minor Subject Thesis, DIKU, University of Copenhagen

Grant H., Lai C. K., (1998) Simulation modelling with artificial reality technology (SMART): an integration of virtual reality and simulation modelling, Proceedings of the 30th conference on Winter simulation, Washington, D.C., USA, IEEE Computer Society Press

Hanne K.H. (1992) Multimodal Communication, Natural Language and Direct Manipulation (Gestures) in Human Computer Interaction. In Edwards A.D.N. and Holland S. (eds.) Multimedia Interface Design in Education, NATO ASI Series, Vol. F 76, © Springer-Verlag Berlin Heidelberg

Hasset J., (1978) A primer of psychophysiology, W.H. Freeman and Co., San Fransisco

Herpers R., MacLean W. J., Pantofaru C., Wood L., Derpanis K., Topalovic D., Tsotsos J. (2001) Fast hand Gesture recognition for real-time teleconferencing applications, IEEE proceedings of the Second International Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real Time Systems, RATFG-RTS'01, Vancouver Canada, IEEE Computer Society

Hart, S.G., (1987) Background Description and Application of the NASA Task Load Index (TLX), Proceedings of the Department of Defence Human Engineering Technical Advisory Group Workshop on Workload. (NUSC6688) Newport, RI: Naval Underwater Systems Center, pages 90-97.

Holzman P.S., Levy D.L. (1977) Smooth pursuit eye movements and functional psychoses: A review. Schizophr. Bull 3

Howell J. A., Buxton H., (2002), Visually Mediated Interaction Using Learnt Gestures and Camera Control, In Wachsmuth I and Sowa T., (eds) Gesture and Sign Language in Human-Computer Interaction, International Gesture Workshop, GW 2001, Springer Lecture Notes in Computer Science, Vol. 2298, Springer-Verlag, Heidelberg, Germany

HP invent – Hewlett-Packard Company, Palo Alto, California, USA

Hutchinson T.E., White K.P., Martin W.N., Reichbert K.C, Frey L.S (1989) Human - computer interaction using eye-gaze input. IEEE Transactions. Syst. Man. Cybern.

IBM – International Business Machines Corporation, Armonk, New York, USA

Immersion, San Jose, California, USA

Instance H., Howarth P., (1994) Keeping an eye on your interface: the potential for eye-gaze control of graphical user interfaces, Proceedings of HCI'94.

Ishii H., Wisneski C., Brave S., Dahley A., Gorbet M., Ullmer B., Yarin P. (1998) ambientROOM: Integrating Ambient Media with Architectural Space, CHI 98 conference summary on Human factors in computing systems, Los Angeles, California, USA

Ishii H., Ullmer B., (1997) Tangible bits: towards seamless interfaces between people, bits and atoms, Proceedings of the SIGCHI conference on Human factors in computing systems, Atlanta, Georgia, United States

Jacob R.J..K. (1991) The use of eye movements in human-computer interaction techniques: what you look at is what you get. ACM Transactions on Information Systems, Vol 9, No.3, April 1991

Jantti V., Lang A.H.,Keskinen E., Lehtinen I., Pakkanen A. (1983) Acute effects of alchol on saccadic eye movements and subjective evaluations of intoxication. Psychopharmacology 79

Karn K. S., (2002) Definitions "dwell" vs. "gaze", Eye-Movement mailing list, http://www.jiscmail.ac.uk/files/eye-movement/introduction.html, 3rd September 2002

Kendon A. (1992) The negotiation of context in face-to-face Interaction. In Duranti and Goodwin eds., Rethinking context: Language as an interactive phenomenon. Cambridge, UK: Cambridge University Press

Kennard C., Crawford , T.J, Henderson L. (1994) A pathophysiological approach to saccadic eye movements in neurological and psychiatric disease. Journal of Neurology Neurosurgery Psychiatry 57

Lakoff G., Johnson M., (1980) Metaphors we live by, The University of Chicago Press, Chicago, USA

Lankford C., (2000) Gazetracker: Software Designed to Facilitate Eye Movement Analysis, Eye Tracking Research & Application Symposium 2000, Palm Beach Florida, USA, ACM press

Lankford C., (2000) Effective EyeGaze Input Into Windows, Eye Tracking Research & Application Symposium 2000, Palm Beach Florida, USA, ACM press

Laptev I., Lindeberg T., (2001) Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features, in Kerckhove M. (Ed.) Scale-Space and Morphology in Computer Vision, Proceedings of Third International Conference, Scale-Space 2001, Vancouver, Canada, Springer Lecture Notes in Computer Science, Vol. 2106, Springer-Verlag, Heidelberg, Germany

LC-Technologies Inc., 9455 Silver King Court, Fairfax, Virginia 22031-4713 U.S.A, http://www.eyegaze.com/

Lee G. A., Kim G. J., Park C.M., (2002) Modeling virtual object behavior within virtual environment, Proceedings of the ACM symposium on Virtual reality software and technology, Hong Kong, China, ACM Press

Lepouras G., Eir G. R. S., (1999) It's not Greek to me, SIGCHI Bulletin Vol31, Num. 2, ACM Press

Lewis O.J. (1989) Functional Morphology of the Evolving Hand and Foot. Oxford: Clarendon Press

Lord of The Rings Trilogy (2001-2003) New Line Cinema

Lotus Software, IBM software Group, Cambridge, Massachusetts, USA http://www.lotus.com

Madsen K.H., (1994) A Guide to Metaphorical Design, Communications of the ACM, Volume 37, Issue 12

Marcel S., Bernier O., (1999) Hand Posture Recognition in a body-Face Centered Space, In Braffort A., Gherbi R., Gibet S., Richardson J., Teil D. (Eds.), Gesture-Based Communication in Human-Computer Interaction, Proceedings of International Gesture Workshop, GW'99, Gif-Sur-Yvette, France, Springer Lecture Notes in Computer Science, Vol. 1739, Springer-Verlag ,Heidelberg, Germany

Marrin T. and Paradiso J. (1997) The Digital Baton: a Versatile Performance Instrument, International Computer Music Conference, Thessaloniki, Greece, September 1997

McGee D. R., Cohen P. R., Wu L., (2000) Something from nothing: augmenting a paper-based work practice via multimodal interaction, Proceedings of DARE 2000 on Designing augmented reality environments, Elsinore, Denmark, ACM Press

Merriam-Webster Dictionary, http://www.m-w.com/home.htm

Microsoft Corporation, Redmond, Washington, USA, http://www.microsoft.com

Minority Report (2002) 20th Century Fox, DreamWorks

Mulder A. (1994) Virtual musical instruments: accessing the sound synthesis universe as a performer, School of Kinesiology, Simon Fraser University, Burnaby, Canada

Müller, P. U., Cavegn, D., d'Ydewalle, G. & Groner, R. (1993). A comparison of a new limbus tracker, corneal reflection technique, purkinje eye tracking and electro-oculography, *in* G. d'Ydewalle & J. V. Rensbergen, eds, `Perception and Cognition', Elsevier Science Publishers

Mc Neil D. (1992) Hand and Mind: What Gestures reveal about Thought, University of Chicago Press, Chicago

Napier J. (1980) Hands, Pantheon Books, New York

Nolker C., Ritter H., (1999), GREFIT: Visual Recognition of Hand Postures, In Braffort A., Gherbi R., Gibet S., Richardson J., Teil D. (Eds.), Gesture-Based Communication in Human-Computer Interaction, Proceedings of International Gesture Workshop, GW'99, Gif-Sur-Yvette, France, Springer Lecture Notes in Computer Science, Vol. 1739, Springer-Verlag, Heidelberg, Germany

Patmore D.W., Knapp R.B., (1998) Towards an EOG-Based Eye Tracker for Computer Control, Proceedings of the third International ACM Conference on Assistive technologies ASSETS 1998, Marina del Rey, California USA, ACM Press

Pirhonen A., Brewster S., (2001) Metaphors and Imitation, Integrating Metaphors, Multimodality and Multimedia Workshop Proceedings, held at PC-HCI 2001, Patras Greece

Pynadath D.V., Tan H.Z. Horowitz D.M. (1993) A study of the information capacity of human eye movement for augmentative communication. RESNA'93 13

Quek F.K.H. (1996) Unencumbered Gestural Interaction, IEEE Multimedia, Winter 1996.

Ramloll R., (2000) Supporting Co-operative Work through Ubiquitous Awareness-Filtration Mechanisms, PhD Thesis, Lancaster University

Ramage, M., (1999) The Learning way: Evaluating Co-operative Systems, PhD Thesis, Lancaster

Rubine D., (1992) Combining Gestures and Direct manipulation, Proceedings of the SIGCHI conference on Human factors in computing systems, May 3-7, 1992

Sallnäs E. L., Rassmus-Gröhn K., Sjöström C., (2000) Supporting presence in collaborative environments by haptic force feedback, ACM Transactions on Computer-Human Interaction, Vol 7, Num. 4, ACM Press

Salvucci D.D. (1999) Inferring Intent in Eye-Based Interfaces: Tracing Eye movements with Process models, Proceedings of the SIGCHI conference on Human factors in computing systems, Pittsburgh, Pennsylvania, USA, ACM Press

SensAble Technologies, Inc., Woburn Massachusetts, USA http://www.sensable.com

Sibert L.E. and Jacob R.J.K (2000) Evaluation of Gaze Interaction Proceedings of the SIGCHI conference on Human factors in computing systems, The Hague, The Netherlands, ACM Press

SICS – Swidish Institute of Computer Science, Kista Sweden, http://www.sics.se/dive/dive.html

SignPS, http://www.techinfo.rwth-aachen.de/Forschung/SLR/VGErkennung/signps.html

SmartKom Consortium, http://smartkom.dfki.de/

Sommerlad, E. (1992) A Guide to Local evaluation. Evaluation Development and Review Unit, Tavistock Institute of Human Relations, London

Sony Corporation, Tokyo Japan http://www.sony.net/

Sony Ericssson, London, UK http://www.sonyericsson.com

Sowa T., Frohlich M., Latoschik M.E.(1999) Temporal Symbolic Integration Applied to a Multimodal System Using Gestures and Speech, In Braffort A., Gherbi R., Gibet S., Richardson J., Teil D. (Eds.), Gesture-Based Communication in Human-Computer Interaction, Proceedings of International Gesture Workshop, GW'99, Gif-Sur-Yvette, France, Springer Lecture Notes in Computer Science, Vol. 1739, Springer-Verlag, Heidelberg, Germany

Starker I, Bolt R.A. (1990) A gaze-responsive self-disclosing display, Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press

Stern J., Walrath L.C., Goldstein R. (1984) The endogenous eyeblink, phychophysiology 21, 22-33

Stiefelhagen R., Zhu J., (2002), Head Orientation and Gaze Direction in Meetings, Proceedings of the SIGCHI conference on Human factors in computing systems, Minneapolis, Minnesota, USA, ACM Press

Tanriverdi V., Jacob R. J. K., (2000) Interacting with eye movements in virtual environments, Proceedings of the SIGCHI conference on Human factors in computing systems, The Hague, The Netherlands, ACM Press

Tecce J.J., Gips J., Olivieri P.C., Pok L.J., Consiglio M.R., (1998), Eye Movement control of computer functions, International journal of Psychophysiology, vol.29, no. 3

Thorisson K.R. (1998) Real-Time Decision-Making In Multimodal Face-To-Face Communication, ACM conference Autonomous Agents 98, Minneapolis MN, USA

Triesh J., Wienghardt, Mael E., von der Malsburg C. (1999) Towards Imitation Learning of grasping movements by an Autonomous Robot, In Braffort A., Gherbi R., Gibet S., Richardson J., Teil D. (Eds.), Gesture-Based Communication in Human-Computer Interaction, Proceedings of International Gesture Workshop, GW'99, Gif-Sur-Yvette, France, Springer Lecture Notes in Computer Science, Vol. 1739, Springer-Verlag, Heidelberg, Germany

UCL – University College London, Department of Computer Science, Virtual Environments & Computer Science Group, ReaCTor System
http://www.cs.ucl.ac.uk/research/vr/Projects/Cave/

Vertegaal R., (1999) The Gaze Groupware System: Mediating Joint Attention in Multiparty Communication and Collaboration, Proceedings of the SIGCHI conference on Human factors in computing systems, Pittsburgh, Pennsylvania, USA, ACM Press

Vertegaal R., Slagter R., van der Veer G., Nijholt A., (2001) Proceedings of the SIGCHI conference on Human factors in computing systems, Seattle, Washington, USA, ACM Press

Viveash J.P., Belyavin A.J, Waters M., Stott J.R.R . (1996) The accuracy of eye movement measurement under operational conditions, Journal of Defence Science, Vol.1

Ware C., Mikaelian H.H. (1987) An evaluation of an eye tracker as a device for computer input, ACM Conference on Human Factors in Computing Systems and Graphics Interface, Toronto, Canada

Washburn S.L., (1960) Tools and Human Evolution, Scientific American, Vol. 203, Issue 3

Willier A., Marque C., (2002) Juggling Gestures Analysis for Music Control, In Wachsmuth I and Sowa T., (eds) Gestrure and Sign Language in Human-Computer Interaction, International Gesture Workshop, GW 2001, Springer Lecture Notes in Computer Science, Vol. 2298, Springer-Verlag , Heidelberg, Germany

Wenzel E.M (1992) Three - Dimensional Virtual Acoustic Displays In Blattner M and Dannenberg R (eds.). Multimedia Interface Design, ACM Press frontier series, New York

Wexelblat A. An Approach to natural gesture in virtual environments. In ACM Transactions Computer- Human Interaction 2,3, September, 1995

Wilson F. (1998) The Hand, Pantheon Books, New York

Xerox Corporation, Stamford, USA, http://www.xerox.com

Young L.R., Sheena D. (1975) Survey of eye movement recording methods. Behavior Research Methods and Instrumentation 7(5)

Yu R. W., Ramloll R., Brewster S.A., (2000) Haptic Graphs for blind computer users, First International workshop on Haptic Human-Computer Interaction, University of Glasgow, UK

Zhai S., Morimoto C., Ihde S., (1999) Manual and Gaze Input Cascade (MAGIC) Pointing, Proceedings of the SIGCHI conference on Human factors in computing systems, Pittsburgh, Pennsylvania, USA, ACM Press

5th Glove'95 (1996) User's manual

5DT, 25 De Havilland Crescent, Persequor Park, 0020, South Africa, http://www.5dt.com/

Appendix A – Interaction Techniques

This appendix first provides an overview of the commonly used HCI interaction styles, and then discusses their disadvantages. Some of the techniques been used for many years and continue to be used, some are relatively more recent and some are still evolving. The positive and negative aspects of each method are highlighted and examples of each are presented.

Dix et al. (2004) have identified the following key methods of interacting with computers:

- Command line interface
- Menus
- Natural language
- Question/answer and query dialog
- Form-fills and spreadsheets
- The WIMP interface
- Point-and-click interfaces
- Three-dimensional interfaces

These interaction methods will be briefly discussed in the following sections.

A.1 Command line interface

The command line interface was the first method of computer-user interaction. Familiar examples of command line interfaces include MS-DOS (Microsoft Corp.) (Figure A.1) and UNIX environments. The user expresses the commands to the computer directly, using a set of predefined commands (for example, *cd* to change the current directory), and function keys. The command line is still the only way of interacting with the computer in some systems, while in others it is a secondary method, which provides quick access to system commands for experienced users, for example the *RUN* command for MS-WINDOWS (Microsoft Corp.). It is also frequently used for remote access, (applications such as telnet) due to the low bandwidth it requires.

Like all interaction styles, command line interfaces have both pros and cons. They provide the users with the ability to directly access system functionality, combine commands or specify (using switches and parameters) how these commands will be performed and on which data. However, in order to be able to effectively employ this flexibility in their interaction, users have to spend time learning the system as typically no semantic cue is provided. As a result, most users of such systems tend to be experts, since many times the commands used are obscure and inconsistent across platforms (for example, *dir* in MS-DOS as shown in figure A.1, and *ls* in UNIX both result in a directory listing).

Figure A. 1 - MS-DOS Command line

A.2 Menus

In menu-driven interfaces the user interacts with the system through a set of options, which are displayed on the screen. The interaction is done either with the use of a mouse or through keystrokes, using function keys, numeric or combinations of function keys and alphanumeric keys. Figure A.2 shows the MS-DOS Editor (Microsoft Corp.), where the menu appears on a menu bar at the top, and the sub menus "drop down" when their "parent" command is invoked. The commands used in menus are usually grouped in a logical hierarchical order. Figure A.1 shows an example of such a grouping, where the file operations are grouped together.

Since the options are visible they require recognition of the command rather than recall. For this reason, menu designers need to pay attention to ensure that that the menus are logical and consistent. There are many cases where problems caused by inconsistency are encountered, for example when translating applications from their original language into another. The translation is done to assist the users to associate better, but in many cases the translation uses words that are either not commonly used or not appropriately used, making the menu difficult to use (Lepouras et al. 1999).

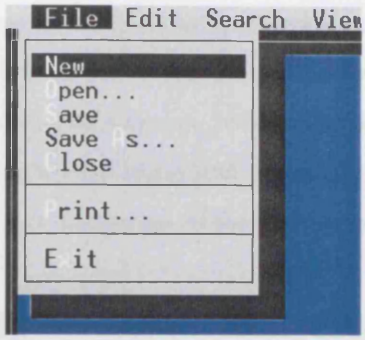


Figure A. 2 - Drop Down menu (MS-DOS Editor, Microsoft Corp.)

From a comparison of menus with command line interfaces, it is evident that the learning overhead required is much less for systems that use menus. This is because the human ability to recall is much inferior to that of recognition based on a visual cue (Dix, 1998). However, this difference between recognition and recall is where problems can occur with menus. As pointed out earlier, the design of a menu is the key to its functionality. In command line interfaces the user may choose to learn the whole range of commands or only a subset of most commonly used ones, and if needed refer to a manual to look up any other needed command. In a menu the user may use only the commands that are included in the menu. So the problem is to identify those items to include in the menu, and how they should be grouped. Including too many items will increase the power of the application, but will potentially result in menus that are too long or too numerous, as can be seen in Discreet 3DstudioMax (Discreet). The other problem is that of grouping, since items that relate to the same topic should be grouped in under the same heading, but some items could be grouped in more than one heading.

A.3 Natural language

Natural language interfaces are appealing to users that are unable to remember a command or get lost in a hierarchy of menus. There has been much research on natural language understanding both for speech and written input. But as language is ambiguous at many levels this makes it very difficult for machine processing. This ambiguity can be found in the syntax or structure of a phrase. Dix (1998) gives an example of such an ambiguity in the phrase "the man hit the boy with the stick" where it is not clear if the boy was hit by the man using the stick or whether the boy had the stick in his possession when he was hit. Another level that ambiguity may occur is in the meaning of words, and that is where the context and our general knowledge help us to overcome any ambiguity. Due to this it seems likely that general-purpose natural language interfaces will not become available soon.

Today we have a number of systems available that can understand specialised subsets of a language. These have been quite successful but as the use of a certain vocabulary and syntax is required most of the time, it is debatable whether they can really be considered to be natural language. A few search engines use natural language processing, but, as mentioned earlier, they are quite limiting and the outcome is not always the expected. Figure A.3 shows the answers that were returned when the question "what is evolution?" was given to the Ask Jeeves search engine (Ask Jeeves). Most suggested links pointed to commercial sites that sell products that are either named or have as part of their name the word "evolution", rather than sites relating to "evolution".

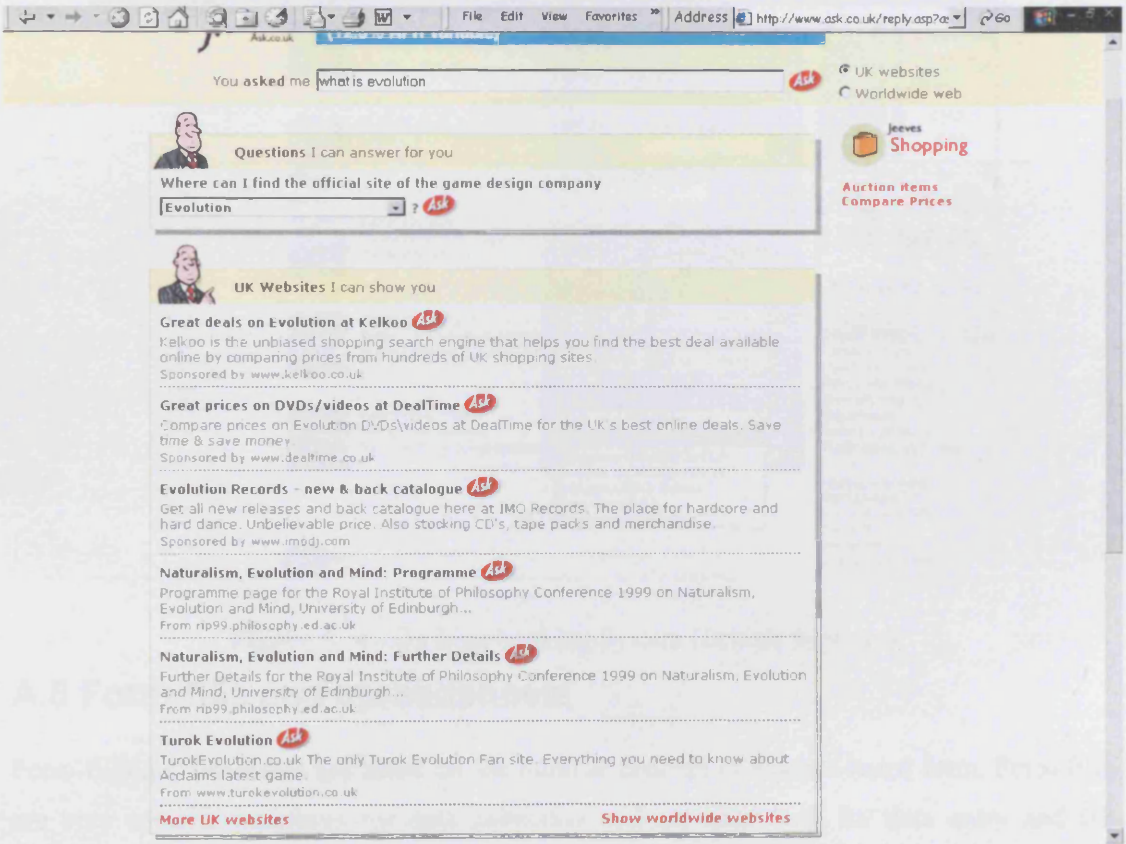


Figure A. 3 - Searching the Web using Natural language

A.4 Question/answer and query dialog

Question /answer dialog interfaces are based on the principle of asking the user a series of simple questions, which will have yes/no or multiple choice answers, to lead him/her through the interaction. These interfaces are easy to use, have a very small learning curve but also have very limited functionality. An example of such a system is an on-line air ticket booking system (figure A.4) where the user may select options such as the type of trip, where it will commence and the final destination and dates of travel.

In contrast to the simplicity of question/answer dialog systems, more complex query languages can be used to construct queries to retrieve information from a database. These queries are natural language style phrases, but are highly structured and very specific.

Effective use of a query language to create a dialog often requires the user to have good knowledge of database systems.

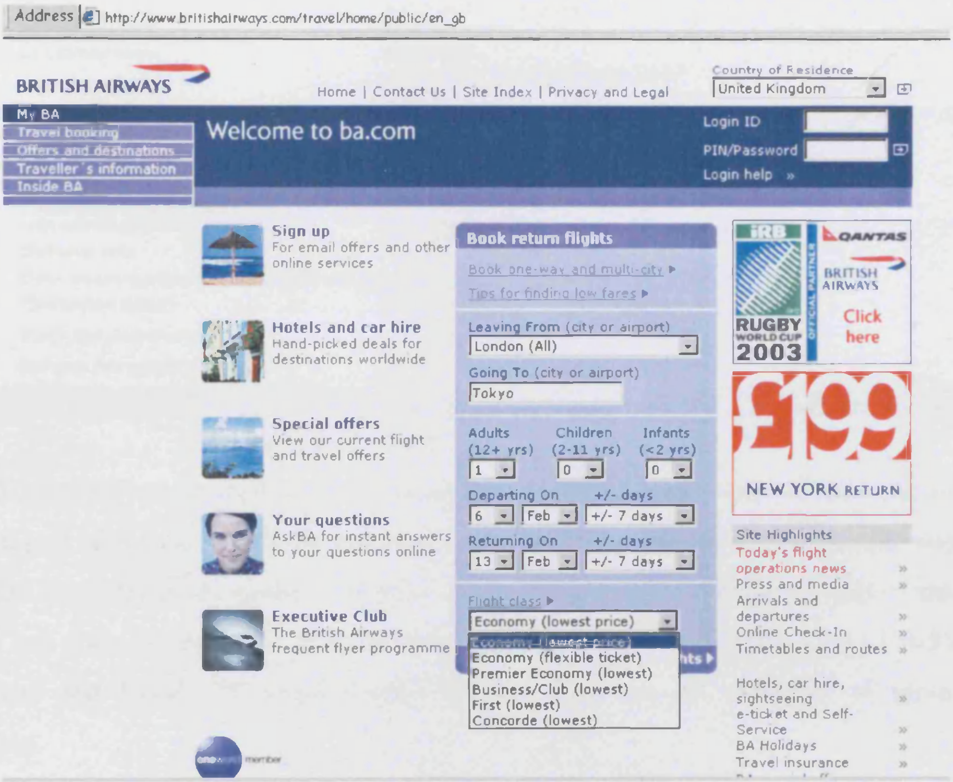


Figure A. 4 - On Line booking System (British Airways)

A.5 Form-fills and spreadsheets

Form-filling applications are based on the familiar concept of a paper-based form. Form-fills are very specific interfaces for data collection and are used both for data entry and for specifying fields in data retrieval. In such systems the user is presented with a display very similar to that of either an existing paper form or an interface that is an electronic implementation of such a system. These electronic forms have boxes that represent fields to be filled or checked. Certain boxes may often be left blank and in the event of a mistake a correction facility is often provided. Interfaces that are used to enter data into large databases, such as those used by utility companies, are usually of this type.

However, users sometimes become frustrated, as data fields that they are called to fill in as mandatory either do not apply to them or, in the user's opinion, require data that is not important. In such cases, the system will not allow the user to proceed to the next step until all data has been completed, while with a paper based form the user would be able to only fill in data that he/she feels is necessary before submitting it. Figure A.5 shows an example of such a form, which is used for the ACM CHI2003 (ACM SIGCHI) conference registration.

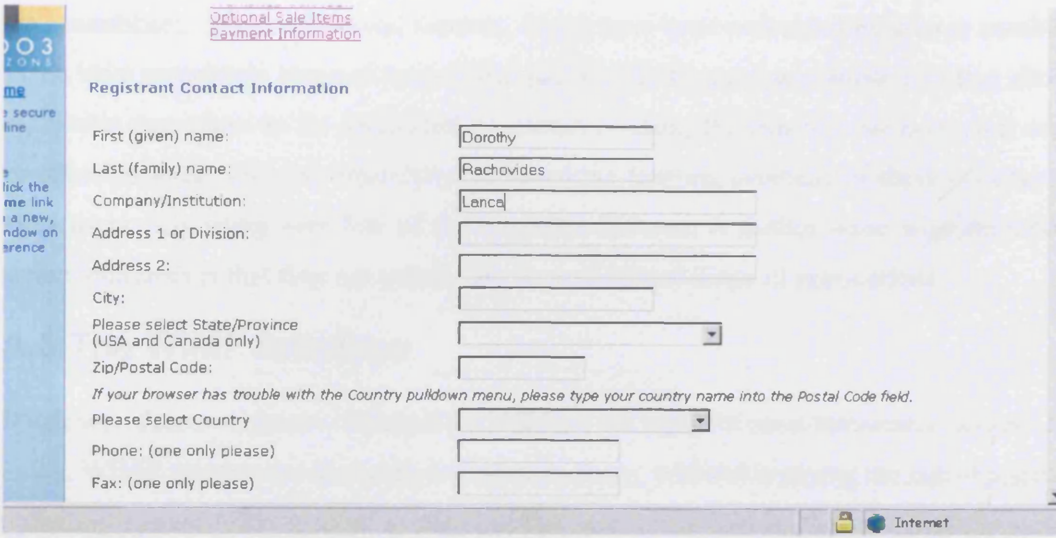


Figure A. 5 - CHI 2003 registration form

Spreadsheets are a more sophisticated variation of form filling. A system of rows and columns forms a grid of cells in which the user may enter numeric or textual data. The user may enter formulae that refer to the contents of other cells, and in that way bridge the gap of input and output and make the interface more flexible and more natural. VisiCalc, Lotus 1-2-3 (Lotus Software) and Excel (Microsoft Corp.) (Figure A.6), are all examples of spreadsheet packages.

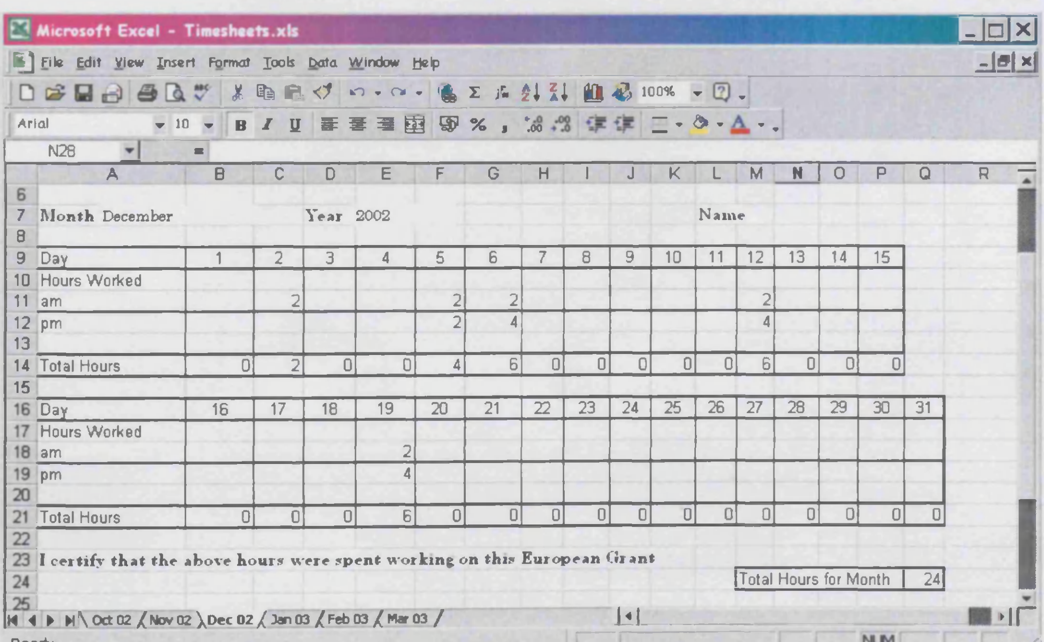


Figure A. 6 - Working on spreadsheets with Microsoft Excel

Spreadsheets that are used today have advanced considerably since the development of VisiCalc. Today's spreadsheets incorporate tools to produce Graphs, to perform Statistical analysis, and to set up a small database. Additionally, by using the more advanced tools provided, the user may develop a form-fill interface through which the data will be entered to the spreadsheet. These additional features, which have been welcomed by a large number of users, have completely changed spreadsheet use. Unlike the original, simple tool that allowed arithmetic operations to be performed on ranges of data, the user is now presented with a complex package. This has drastically increased the learning overhead of these systems with most users only using very few of the available features. A further issue with spreadsheet styled interfaces is that they are suited only for a restricted range of applications.

A.6 The WIMP interface

*W*indows – *I*cons – *M*enus – *P*ointers (WIMP) are the basis for most interactive systems used today. WIMP systems are also called window systems, without implying the use of a specific operating system. This type of system has become a standard in the past decade, as most widely used platforms moved to this type of interaction style. The Apple interfaces, MS Windows and some interfaces for Unix flavours (XWINDOWS and IRIX) belong to this category. An example of such an application is shown in figure A.7, where a typical WIMP package, Macromedia Fireworks, is used to create a poster.

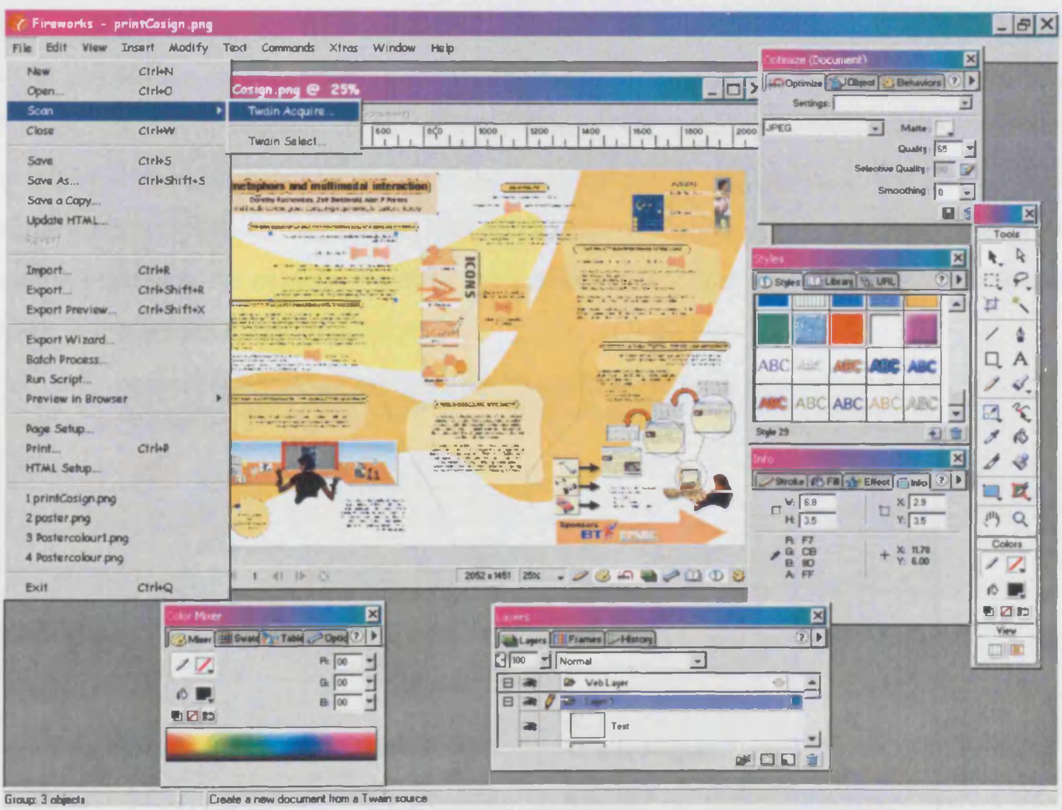


Figure A. 7 - Using Macromedia Fireworks to create a poster

The ability to run a variety of applications in different windows on the same screen and share data between these applications is one of the major benefits of WIMP interfaces. The learning overhead is much smaller than that of Command line interfaces, and they are less limited than menu systems. However, the user who wants to be able to "administer" his/her WIMP system might get frustrated with the complexity and depth of the menus and the variety of Icons presented. A further problem that any user may face is that of the busy desktop. Overcrowding the desktop with icons and windows may pose difficulties in interaction. In figure A.7, the menus and pallet windows take up more space than the actual document that the user is working on.

A.7 Point-and-click interfaces

Point-and-click systems can be considered as a step further from WIMP interfaces. Interfaces of this type are mainly used in multimedia applications and the web, as they enable the user to interact through a single click. This click can be on a graphical element, a *hotword* (a word that has hypertext linking properties) or a recognizable button. Point-and-click systems are closely related to the hypertext-hypermedia philosophy.

Many applications that implement this type of interaction mechanism are touch sensitive information systems in combination with menu driven interfaces. Info kiosks, cash points, and restaurant order taking systems commonly feature point-and-click interface applications. Figure A.8 is a screen shot from the British Airways Check-in Kiosks (British Airways) found in UK airports for passenger check-in. This particular screen allows the user to select a seat. The seat that is automatically allocated by the system (15C in this example) is indicated on the top left by showing a passenger seated in the graphical representation. When the user "points", by touching the touch-screen, another seat on this representation, it is highlighted (17B in this example) and then the passenger will be represented in that seat. Only the available seats are shown in the graphical representation.

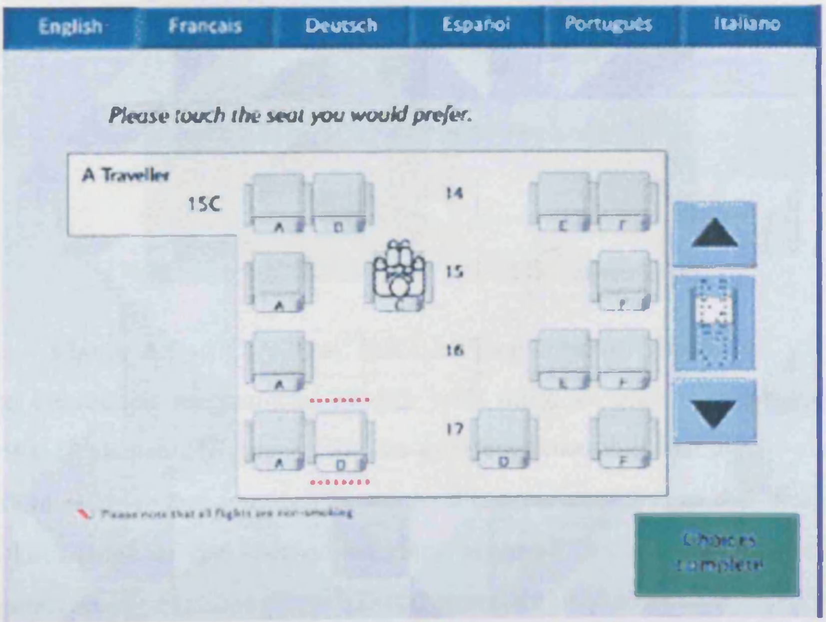


Figure A. 8 - Selecting a seat on a British Airways flight (British Airways)

Point-and-click systems are simple and straightforward to use with a minimum learning overhead as they usually require simple instructions which can be displayed on screen during interaction. However, users of WIMP applications can sometimes have problems in adapting to point-and-click systems that require only a single click to invoke a response. For example, WIMP systems tend to use double clicks, so WIMP users tend to double click with point-and-click systems and this may be translated as two clicks, one on the object that is on the user's screen and one on the link that the user chose to go to. Some systems, such as web browsers, take that into consideration and ignore the second click in a double click, but this is not always the case. Another issue with point-and-click systems is that they are not really suitable for all types of application, being geared towards simple tasks rather than more complex tasks, such as word processing.

A.8 Three-dimensional interfaces

The term *three-dimensional interfaces*, or 3D, covers a wide range of interface, starting from WIMP elements that have a sculptured effect to 3D virtual environments (figure A.9). Interfaces that provide a 3D workspace belong to this interaction style. The objects that are displayed in these systems are usually flat but perspective is used, in such a way that depending on the angle and the distance they shrink to appear "further away". Examples of 3Dsystems include Dive (SICS), COVEN (CVOEN Consortium), VR-VIBE (Benford et al., 95) and UCL's ReaCToR CAVE environment (UCL).

Figure A.9 - UCL's ReaCToR CAVE environment (ReaCToR)

Again these interaction mechanisms overlap with those of others, in particular those of sculptured WIMP elements (figure A.10). The difference though is that the 3D elements invite the user to use reflexes and intuition based on the experience in the real world, instead of relying on knowledge of the specific interface acquired through training. Browsers and computer games are applications where 3D-sculptures are used extensively.

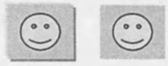


Figure A.10 - a button with a with a 3D effect and with no effect

Three–dimensional environments are more inviting to the user through drawing on the real world experiences that the user may have. This is more evident in virtual environments where the user moves and interacts within a virtual environment and rather than being merely a spectator. This also applies to 3D buttons in WIMP interfaces, as new users can easily identify that a sculptured element is a button. But there are many cases where 3D effects are used excessively, and in these cases any possible sense of differentiation that would have been added by the use of 3D effects on buttons is lost. A further problem with 3D environments, mainly relevant to Virtual environments, is that users can become overloaded with information and get "lost" in the 3D world.

Appendix B – Eye Tracking Techniques

B.1 Techniques based on Reflected Light

The five techniques that comprise this category are all based on the reflection of light on some part of the eye. Most of these techniques use an infrared light beam. These techniques are:

- *Limbus Tracking*

  This technique is based on the reflection of light on the *limbus* of the eye, the boundary between the white sclera and the dark iris of the eye. Due to the contrast in colour between the sclera and the iris, the limbus can be usually be easily optically detected and tracked. This technique tracks the position and shape of the limbus in relation to the head. Therefore, either the head movements must be restricted, or the tracking equipment must be worn on the user's head.

  A problem that may occur with this technique is that the limbus might be partially covered by the eyelids. During a tracking session the user's eyelids may move, which will vary the amount of the limbus that they cover. For this reason Scott and Findlay (1993), regard this technique suitable only for precise horizontal tracking.

- *Pupil Tracking*

  This is a technique very similar to limbus tracking, but in this case the boundaries of the pupil and the iris are detected. As the technique used is again based on tracking the relative position of the pupil boundary to the users head, the user must be either fairly still or wear the equipment on his/her head.

  In this case, the eyelids do not cover as much of the pupil so vertical tracking is possible. Also, as the border of the pupil is typically sharper than that of the limbus, this results to a higher resolution. However, the contrast is lower between the pupil and iris than that of the limbus, making the border detection more difficult

- *Corneal and Pupil reflection relationship – Video Oculography (VOG)*

  When light, typically infrared light, is shone into the user's eye, four reflections occur on the boundaries of the lens and the cornea. This is due to the Perkinje phenomenon in which all of the colours of the spectrum do not fade equally with diminishing light. This phenomenon is actually a shift in the relative brightness of certain colours as illumination diminishes. These images are referred to as the Perkinje images (figure B.1). The first Perkinje image, or *glint* as it is also called, together with the reflection off the retina, or bright-eye as it is also referred to, can be recorded using an infrared sensitive camera. As the user's eye moves horizontally or vertically, the relative positioning of these two images change accordingly. The direction of the user's gaze can be determined by calculating from these relative positions.

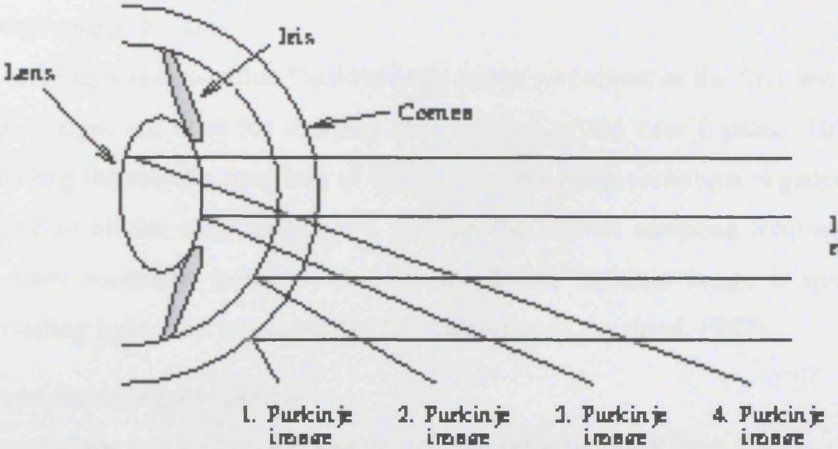


Figure B. 1 Purkinje images (Glenstrup et al., 1995)

The main problem with this technique is that a good view of the eye must be maintained, so the head movements must again be minimal. For this reason, the VOG system is either worn by the user on his/her head (figure B.2) or may require a movement restricting frame.

Figure B. 2 ASL Head mounted pupil-cornea VOG tracker (ASL)

- *Corneal reflection and Eye Image using an Artificial Neural Network*

This technique uses Artificial Neural Networks (ANN) for the computation of the point that the user is gazing. The digitised video images used here are wide-angle, so that an image of the user's entire head is captured. A stationary light is placed in front of the user and the system detects the glint of the right eye. A smaller rectangular part of the video images is then extracted, centred around the glint. This video image, typically 40 by 15 pixels, is the input to the ANN. The output of the ANN is a set of display coordinates.

This techniques has a very long calibration/training time, typically over 30 minutes. However, this procedure is required only on first time use of the system. A further issue is that the accuracy of this technique is 1.5-2°, which is the lowest of all the techniques discussed here. Nevertheless, it does have the advantage that the user may move his/her head more freely (up to 30cm), since an image of the entire head is captured.

- *Purkinje image tracking*

  This technique is also called Dual Perkinje Image technique, as the first, the glint, and the fourth images are used for tracking the direction of the user's gaze. This is done by calculating the relative positions of the two images. This technique is generally the most accurate of all the other techniques, and has the highest sampling frequency (4000Hz). The main constraint is that because the fourth Perkinje image is quite weak, the surrounding light must be controlled (Cleveland and Cleveland, 1992).

- *Infrared Oculography (IROG)*

  This technique is based on the amount of light reflected back from the eye. When a fixed light source is directed at the eye, the amount of light reflected back depends on the eye's position (figure B.3). This has been used for a number of eye trackers, and again the source of light is infrared. Infrared light is preferred in eye tracking because it is "invisible" to the eye, so it does not distract the user, and as infrared detectors are not affected by other light sources, there are no special lighting requirements.

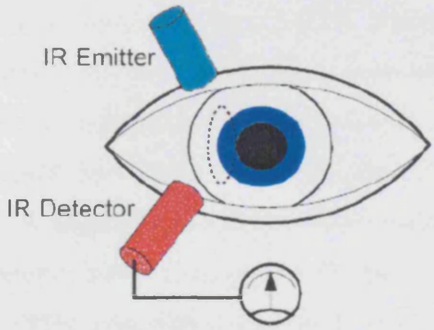


Figure B. 3 - Infrared Oculography (IROG)

Many HCI studies have used the IROG technique (Hutchinson 1989; Starker and Bolt, 1990; Pynadath 1993). The main problem with the technique is eye blinks, as the eye retracts slightly with every blink, which fractionally alters the amount of light reflected for a short time after a blink occurs. A further disadvantage is that it requires a static position of the head, or a head mounted device such as that in figure B.4, and interference can be caused either by scratches on the user's cornea or by the wearing of contact lenses (Young and Sheena 1975).

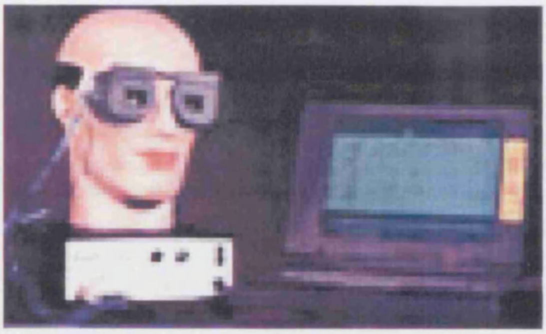


Figure B. 4 EyeTrace 300X IROG (IOTA)

The majority of eye tracking systems today use one of the above techniques. VOG is considered more comfortable, and less intrusive, and for this reason is used also with children. The range of applications that these eye-tracking techniques are used extends beyond HCI. Ergonomics, neurology, ophalmology, sleep disorders, and aviation are areas where this technique is used both in research and in specialised applications.

B.2 Technique based on the electric potential around the eyes

The technique is based on electrooculography, or EOG, which is a direct recording of the electrical potentials generated by eye movement. In saccadic movement of the eye, e.g. when reading, when the eye focuses on a particular spot the potential remains constant. Depending on the direction of the saccadic movement of the eye, the potential will be either positive indicating a left movement, or negative indicating a right movement, as shown in figure B.5 (Hasset, 1978). EOG measured eye movements have been useful in the assessment of cognitive functions (Stern 1984), drug effects (Jantti1983), path physiology (Kennard 1994) and psychiatric disorders (Holzman and Levy 1977) and in the accurate assessment of eye fixation in the operation of an aircraft (Viveash 1996).

Tecce et al conducted two experiments to determine the accuracy of computer control through eye movement using the EOG method (Tecce 1998). Their findings conclude that the EOG method permits successful control of computer functions.

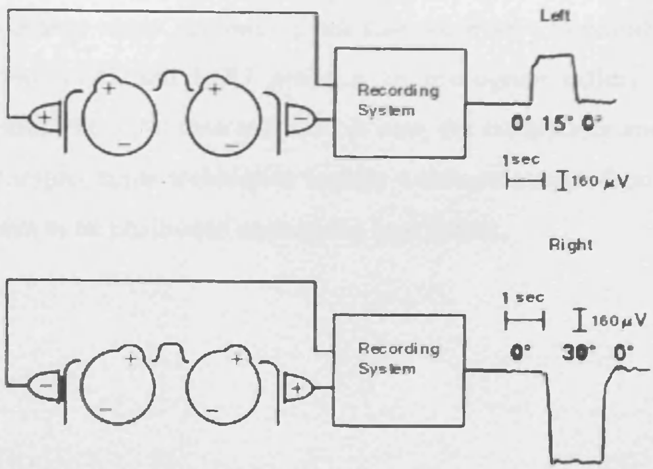


Figure B. 5 : Measurement of electrical potential in saccadic movement of the eyes (Hasset, 1978)

The EOG method has been successfully used by James Gips and Peter Olivieri (1996) in the EagleEyes project, which is reviewed in chapter 3.

B.3 Techniques Based on Contact lenses

It is possible to make reasonably accurate recordings of the direction of focus of the human eyes with the use of special contact lenses. Two methods are used:

In the first method, one or more plane mirror surfaces are engraved on the lenses. Then the reflection of the light beams are used to calculate the position of the user's eye.

In the second method, a tiny induction coil is implanted into the lens, and this allows the recording of the user's eye position through the use of a high-frequency electromagnetic field, which is placed around the user's head.

Both of these methods are problematic, as they are intrusive and may be very uncomfortable for the user, as some available systems have wires connected to the contact lenses, as shown in figure B.6. There are also health issues concerning high-frequency electro-magnetic fields.

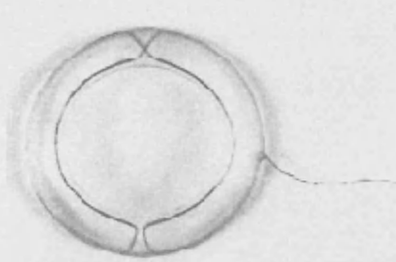


Figure B. 6 Scleral coil (Scalar Medical B.V.)

B.4 Conclusions on Eye Tracking Techniques

All these techniques posses many similarities but also some very important differences, for example, the way that VOG and EOG produce an oculogram differs. In addition, the equipment used to gather the input data can vary in cost, the complexity and how intrusive it is to the user. For example, some techniques require a camera mounted on a monitor, while some require electrodes to be positioned around the user's eyes.

Appendix C - Communicating with gestures

This Appendix examines the role of gestures in face to face communication. It begins by first discussing the evolution of the hand and brain, and goes on to describe how the hand has played a key role in the development of communication

C.1 The evolution of the human hand and brain

A brief discussion is now given of the evolution of the human hand evolution and its role in the development of the brain. This provides insight into how the use of hand movements for survival, grabbing and holding was a driving force in the development of the human brain. In turn, this development of the brain gave the ability of using the hand with dexterity to develop and use tools and communicate with other humans.

Over the past millions of years the human hand has developed into the powerful tool it is now. Ever since our first known ancestor "Lucy" (an Australopithecus who lived about 3.2 million years ago), the hand has been a vital part of our body. Though our ancestors were quite different from us, they performed hand movements similar to those that we carry out in our everyday life. Such movements include scratching, picking, digging, pulling apart, pinching small objects between the thumb and the index finger, grooming and carrying objects in the way we hold and carry a suitcase.

Since our prehistoric ancestors our hand has evolved, providing additional functional advantages (Wilson, 1998):

- The thumb, index and middle fingers can form a *three-jaw-chuck*, which means that the hand can conform to, grasp, and firmly retain irregular solid shapes (such as stones) as shown in figure C.1

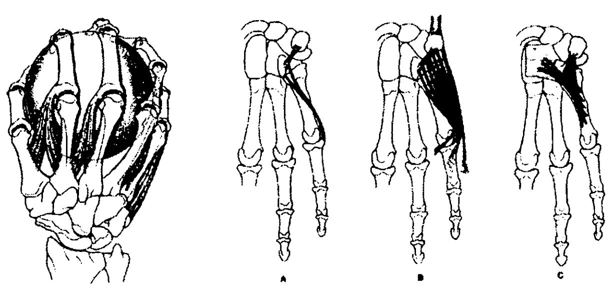


Figure C. 1 - Grip of spherical objects (from Wilson, 1998)

- Finer control can be exerted over objects held between the thumb and tips of the index and middle finger.

- Objects can be held in the hand to perform repeated pounding movements, for example breaking nuts with a rock or digging, because the new wrist structure is able to absorb the shock of repeated hard strikes more effectively than in the ape hand.

There is evidence that large brain size, and in turn brain development, occurred at the same time as tools became more structurally complex, and were kept and transported for long periods. Additionally, complex social structures and language developed gradually in association with the spread of more highly elaborated tool design, manufacture, and use. In comparing the brain sizes of both the chimpanzee and the australopithecines it is found that both weigh approximately 400grm, while the evolved human brain weighs, on average, 1100grm.

Anatomist O.J Lewis explained in 1989: "It is commonly believed that the human hand is essentially primitive; yet the hand has its full quota of features ... which are finely attuned to its specialised role as a delicate manipulative organ ... In the emergent hominids, there must have been progress towards enhancing the overall grasping repertoire of the hand." (Lewis. 1989 p 89).

C.1.1 The evolution of the thumb

The evolution of the hand, from the hand of the Australopithecus to the modern hand was mainly caused by the evolution of the opposable thumb. This was achieved as the thumb grew longer and its attachments to the wrist, and the muscles and tendons moving it, were modified to permit the repositioning of the thumb so that its tip could actually make contact with the tip of each finger and could then reach the other fingertips. Figure C.2 provides an illustration of the movements of the thumb, while Figure C.3 illustrates the structure of the modern human hand.

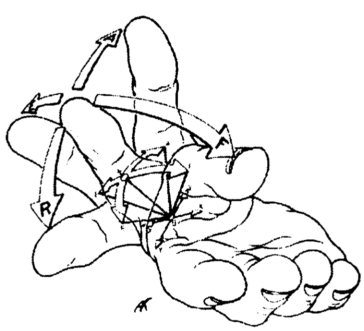


Figure C. 2 - Movements of the thumb (from Wilson, 1998)

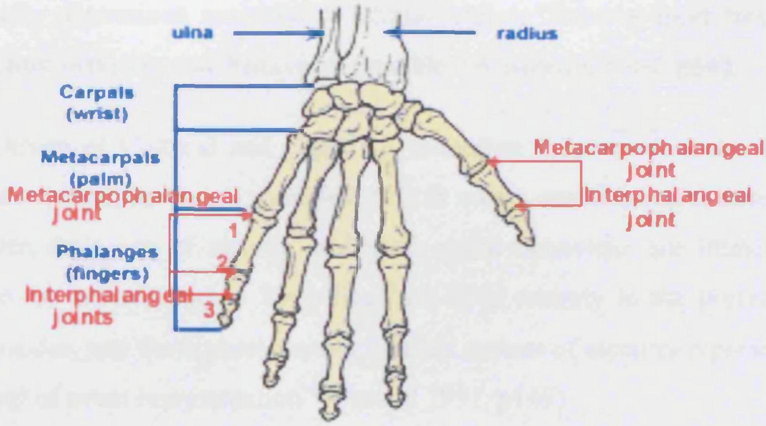


Figure C. 3 - Right hand skeleton, posterior view

It is interesting to consider the role of the thumb in the development of the hand. Wilson refers to the thumb as "The twenty-four Karat Thumb" (Wilson, 1998). This finger is so important, that a hand that has lost the thumb is "at its worst, nothing but an animated fish-slice, and at its best a pair of forceps whose points don't meet properly" (Napier, 1980). As Napier says, "the movement of the thumb underlies all the skilled procedures of which the thumb is capable. ... Without the thumb the hand is put back 60 million years in evolutionary terms to a stage when the thumb had no independent movement and was just another digit."

However, from the humanoid hand not only the thumb evolved. Some time after Lucy, a more mobile joint developed at the base of the small finger. This development in the hominid hand, led to tremendous diversity of dextral skills. Now the human hand, driven by the brain, can make machines that make computers and tools to use with our hands and minds.

C.2 Initiating communication

Anthropologist Peter C. Reynolds (1995) states that stone tool manufacturing was not a task performed individually, as was previously thought. He suggests that complex tools, such as axes and knives may have been customarily manufactured by small groups of people working together, each performing some part of the task. This co-operative effort would have required a means of communication, which would have taken the form of hand signals and other bodily gestures, or vocalisations, or both. So, cooperative tool manufacturing could have been a crucial factor in the evolution of language. An emerging language based in the growth of co-operative tool manufacturing would have fostered the evolution not only of more sophisticated tool manufacturing, but also a more complex social culture and a more refined language. These two patterns of behaviour, both independent, but also in collaboration, would also have been capable of gaining an enhanced representation in the brain. Referring to what is fixed in our genetically determined anatomy, Washburn states, "From a short term point of view, human structure makes human behaviour possible"(Washburn, 1960, p64).

Donald's Theory of Cultural and cognitive Evolution proposes that the behaviour of pre hominid apes and hominids "... complex as it is seems unreflective, concrete, and situation bound. Even their use of signing and their social behaviour are immediate, short-term responses to the environment... Their lives are lived entirely in the present, as a series of concrete episodes, and the highest element in their system of memory representation seems to be at the level of event representation" (Donald 1991, p149)

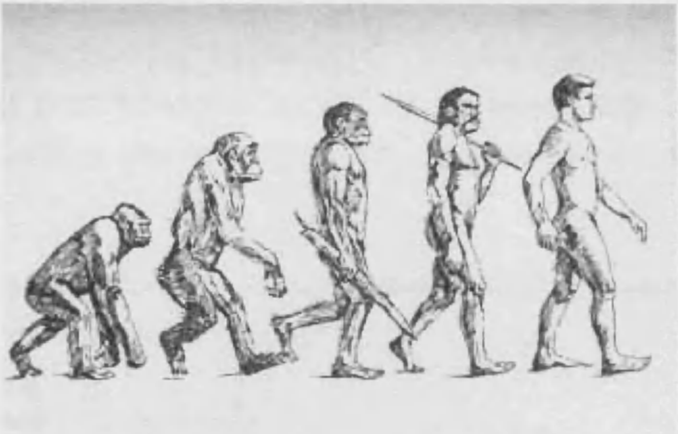


Figure C. 4 - The evolution from Ape to the Modern Man

When mankind evolved to the Homo Erectus, prehistoric man, who could now stand and make tools, lived in small communities and hunted for food, a means of communication was required to support habitual needs. Figure C.4 shows the evolution from the Australopithecus to the Homo Sapiens, the modern man. Donald calls this culture of the erectus a mimetic culture, which was based on the mimetic skill:

> "Mimetic skill or mimesis rests on the ability to produce, self – initiated, representational acts that are intentional, but not linguistic. Mimesis is fundamentally different from imitation and mimicry in that it involves invention of intentional representations... Mimetic skill in the sharing of knowledge without every member of the group having to reinvent that knowledge... The primary form of mimetic expression was and continues to be visio-motor. The mimetic skills basic to child–rearing, tool making, cooperative gathering and hunting, the sharing of food and other resources, finding, constructing and sharing shelter, and expressing social hierarchies and custom would have involved visio-motor behaviour." (Donald 1991,pp169-177)

The One inference that results from Donald's theory is that gestural language was almost certainly employed in communication, and perhaps even the "precursor of the more advanced semiotic inventions underlying speech" (Donald 1991, p220)

C.2.1 The hand as a communication tool

Raising our hand when we are standing in front of a request bus stop will help us get on the bus. Hand movement is but one of the gestures that are incorporated in our everyday life as part of our communication framework. Both of the definitions that follow emphasise that the purpose of the movement or posture is to communicate. In the "Dictionary of worldwide gestures", Buaml and Bauml define a gesture as "a posture or movement of the body or any of its members, that is understood to be meaningful" (Bauml and Bauml 1997, p2). A very similar definition is given by Argyle, "By 'gestures' are usually meant voluntary bodily actions, by hands, head, or other parts of the body, which are intended to communicate" (Argyle1996, p188)

Gestures account for a significant part of our day-to-day lives. The context in which people use gestures varies, but includes the following:

- An accompaniment to normal speech
- A substitute for a foreign language. For example, communication through signs was extensively practised by the Plains Indians to overcome the variety of languages and dialects among their nations.
- A substitute in situations where normal speech becomes inaudible, disadvantageous or dangerous.
- An accompaniment to certain professional activities, e.g. by actors, dancers and practical speakers, to supplement or replace the spoken word.

C.2.2 From thought to gesture

In a simple mind game such as "scissors-paper-rock"(figure C.5) we represent each of the three objects in our mind with a gesture.

Figure C. 5 - Mind Games by Tim Ellis

Gestures exhibit images that cannot always be expressed in speech, as well as images the speaker thinks are concealed. Speech and gesture usually co-operate in expressing meaning. A typical example of this is when someone describes the size of a box and instead of giving exact measurements, gestures as if the box is within reach.

McNeil (1992) classifies gestures in five categories, according to their use and way of delivery:

Iconics: These are gestures that bear close formal relationship to the semantic content of the speech.

Metaphorics: These gestures are similar to iconic gestures in that they are pictorial, but in this case the pictorial content presents an abstract idea instead of a concrete object or event.

Figure C. 6 - Politician using beats (Efron, 1972)

Beats: These gestures are named so because they look like beating music time. The hand moves along with the rhythm of speech (figure C.6). They reveal the speaker's conception of the narrative dialogue as a total. A beat emphasizes the word or phrase it accompanies, not for its semantic value, but at the specific point that it occurs in a dialogue.

Cohesive: The cohesive gestures are quite eclectic in form. They can consist of iconics, metaphorics, pointing gestures, or even beats. Cohesive beats are the gestures that are demonstrated by politicians. Gestural cohesion depends on the repetition of the same gesture form, movement, or locus in the gesture space. Through repetition, the gesture shows the recurrence or continuation of the theme.

Deictics: Pointing has the obvious function of indicating objects and events in the concrete world, but it also plays a part even where there is nothing objectively present to point at. Most pointing gestures in narratives and conversations are at this abstract kind.

C.2.3 Articulating with a gesture

Kendon's "Continuum of Gestures" (McNeil, 1992) shown in Figure C.7, shows the process of articulating a word or concept through a gesture.

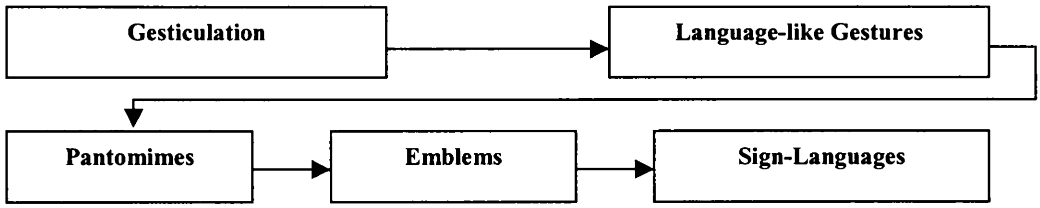


Figure C. 7- "Continium of Gestures" (from McNeil, 1992)

Kendon's continuum is important for distinguishing gestures of fundamentally different kinds. McNeill uses the term "gesture" specifically to refer to the leftmost *gesticulation* end of the spectrum.

In the *gesticulation* sense, gestures are idiosyncratic movements of the hands and arms accompanying speech. These almost never occur in the absence of speech.

Language-like gestures are similar in form but are grammatically integrated in the utterance.

In *pantomime*, (figure C.8) the hands represent objects or actions, but speech is not obligatory. The fading of speech brings pantomime in the middle of Kendon's continuum. In a pantomime there can be either silence or inarticulate onomatopoetic sound effects, like "ssh!", "Click", etc. Successive pantomimes can create sequence-like demonstrations and this is different from gesticulation where successive gestures do not combine.

Figure C. 8 - Gesture for eating (from Efron, 1972)

Emblems are the "Italianate" gestures, mostly insults (figure C.9), but some of them praise, and virtually all of them represent an attempt to exert control over another person. Emblems have standards of well formedness, a crucial language-like property that gesticulation and pantomime lack.

The gestures that Efron refers to are *emblems*. In fig C.9, some of the gestures he considers are presented. These are mainly "Italianate", in their original form, as demonstrated by Southern Italians.

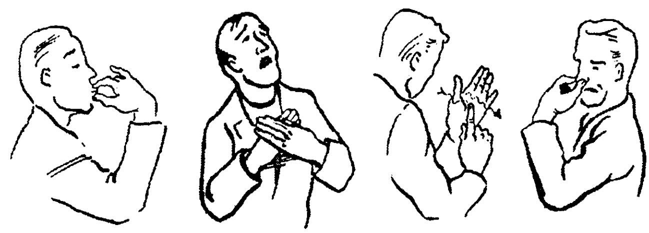


Figure C. 9 - Italianate Gestures (from Efron, 1972)

Emblems are usually used in the absence of speech. This is probably the reason that they exist, since they offer a way of avoiding speech taboos.

Sign languages are full-fledged linguistic systems, with a lexicon, a distinct syntax, arbitrariness, standards of well formedness, and a community of users.

These distinctions of different communicational manual actions are crucial. Different types of movement bear different logical and behavioural relations to speech and are affected differently following intellectual breakdowns. So, highly codified sign languages, such as ASL may be disrupted in ways that result in sign aphasias, very similar to the aphasias of speaking patients. Emblems and pantomimes might replace or supplement to the extent that linguistic capacities can be exploited through another channel. Production of emblems and pantomimes, which are often tested in an attempt to quantify gestural skills, do not vary systematically with the type of aphasia, but seem to be related to the overall severity of the communication insufficiency. Gesticulation bears a more complex relationship to speech, and varies in subtle and intricate ways in relation to the speech it accompanies.

Appendix D - NASA Task Load Index

The NASA Task Load Index (Hart, 1987) is a multi-dimensional rating procedure that provides an overall workload score based on a weighted average of ratings on six subscales: Mental Demands, Physical Demands, Temporal Demands, Own Performance, Effort, and Frustration. A definition of each subscale is provided in table D.1. In practice interface evaluators find that analysis based on the subscale values individually can provide considerable early insight into the suitability of proposed designs.

TITLE	*ENDPOINTS*	*DESCRIPTIONS*
MENTAL DEMAND	*Low/High*	How much mental and perceptual activity was required (e.g. thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?
PHYSICAL DEMAND	*Low/High*	How much physical activity was required (e.g. pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?
TEMPORAL DEMAND	*Low/High*	How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?
EFFORT	*Low/High*	How hard did you have to work (mentally and physically) to accomplish your level of performance?
PERFORMANCE	*Good/Poor*	How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?
FRUSTRATION LEVEL	*Low/High*	How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

Table D. 1 - The Rating Scale Definitions of the standard NASA Task Load Index

Table D.2 shows a questionnaire given to experiment subjects so that they can express their “feelings” about each subscale relevant to the calculation of subjective workload. The lower the values on each scale the lower the workload. An important point to note here is that the scale of the Performance variable has been reversed so that a low value on the Performance scale will contribute to a low workload.

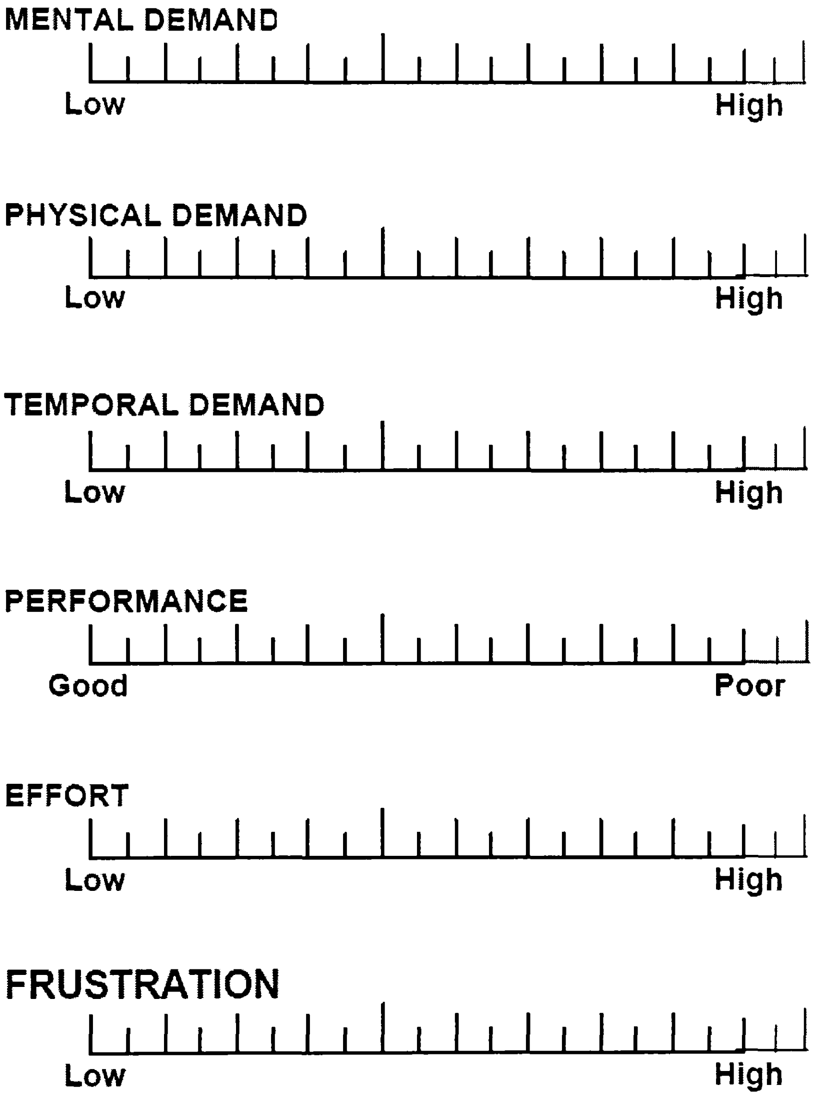


Table D. 2 - NASA TLX Questionnaire

Appendix E - Questionnaires

Interaction Method Questionnaire

The questions in this questionnaire relate to only stage 1 of the evaluation

The Interaction Method

1. Compared to creating the presentation with PowerPoint, would you say that using the Presentation Conductor is:

    ☐ Difficult

    ☐ More complicated

    ☐ Easier

    ☐ The same

    ☐ Extremely easier

2. Please give any other comments on the use of the Presentation Conductor in comparison to the use of PowerPoint:

3. Do you find it easier to interact with the computer using the Conductor Interaction Method, than with existing interaction techniques?

    ☐ Much easier

    ☐ Easier

    ☐ Insignificant difference

    ☐ Difficult

    ☐ Very difficult

4. Did you find it easy to learn to use the method compared to learning to use other interaction techniques

    ☐ Difficult

    ☐ More complicated

    ☐ Easier

    ☐ The same

    ☐ Extremely easier

5. Do you prefer to use such an approach to existing ones?

- ☐ Yes
- ☐ Maybe
- ☐ No

6. Do you find that this type of interaction to be appropriate for this type of application?

- ☐ Very appropriate
- ☐ Appropriate
- ☐ Indifferent
- ☐ Inappropriate
- ☐ Completely inappropriate

7. Do you think there are other applications in which such an interaction method would be beneficial? If so, specify them

The Interaction

8. Was the use of two different modalities, gaze and gesture, natural?

- ☐ it was unnatural
- ☐ no difference
- ☐ it was natural

9. Did you find that the use of gaze was easy and intuitive or were you very conscious at all times of where you were looking at?

- ☐ I was very conscious through out the experiment
- ☐ I was more relaxed after a few minutes
- ☐ I was relaxed from the beginning and found it easy and intuitive

10. When you were selecting by gaze, did you have difficulties in selecting the correct object?

- ☐ I had problems through out the experiment
- ☐ I had problems at the beginning, but improved after a while
- ☐ I had no problems at all

11. Most of the interaction was done through gestures. Did you find the association of the gestures with tasks to be performed:

- ☐ Intuitive
- ☐ Irrelevant to each other
- ☐ Associated in some way but not intuitive

12. Did you find that you were aware at all times what you were doing with your hands, even when gesturing was not taken as input?

- ☐ Yes
- ☐ No
- ☐ Sometimes

13. Did you find the use of gestures in the context of the Presentation Conductor a natural way to interact?

- ☐ it was odd
- ☐ it was unnatural
- ☐ no difference
- ☐ it was natural

14. Did you find that the images used in the gallery icons were representative of the type of media that they were associated with?

- ☐ I could easily associate the icons to the media
- ☐ I found some difficulty, but I understand the association
- ☐ The icons are completely irrelevant with the media they are associated with

15. Was the use of the stage metaphor useful in understanding their functions?

- ☐ Very useful
- ☐ Useful
- ☐ No difference to a toolbar and a window work area
- ☐ Confusing
- ☐ Very confusing

Technology Questionnaire

The questions in this questionnaire relate to both stages of the evaluation

1. Did you find that the use of a small screen and sitting in a constrained area had an effect on how you interacted?

    - ☐ Yes it did constrain me, I felt uncomfortable and needed more space to perform the gestures.
    - ☐ It was ok. I managed to do most of the interactions well
    - ☐ I moved a lot and the gaze did not work for me
    - ☐ I had no problems at all. I would have not needed more space.

2. Was it easy to calibrate the eye tracker for you?

    - ☐ Yes, it calibrated with the first attempt
    - ☐ No, it took more than three attempts
    - ☐ It calibrated with the second attempt
    - ☐ It did not calibrate for me

3. Did you find the use of the eye tracker uncomfortable?

    - ☐ Very uncomfortable
    - ☐ Uncomfortable
    - ☐ ok
    - ☐ Comfortable

4. Were your eyes feeling tired and watery by the end of the session?

    - ☐ Yes, They were watery at times
    - ☐ Relatively tired
    - ☐ No, I didn't feel any tiredness at all

5. Did you find using the data gloves difficult?

    - ☐ Very difficult
    - ☐ Difficult
    - ☐ OK
    - ☐ Easy
    - ☐ Very easy

6. Were the data gloves heavy?

- ☐ Yes
- ☐ No
- ☐ Felt heavy after a few minutes of wearing them

7. How did they fit on your hand?

- ☐ They fitted fine
- ☐ They were far too big for my hand and did not work for me
- ☐ They were too big for my hand, but managed to work most of the time
- ☐ they were too small for my hands

8. Did your hands get tired by the end of the session?

- ☐ Yes
- ☐ No
- ☐ A bit

The questions below relate to stage 2 of the evaluation

9. Did you find that the use of a big screen with the combination of gestures had an effect on how you interacted?

- ☐ I had to adjust my position to be in the range of the Infrared tower quite frequently.
- ☐ I had plenty of space and performed the gestures easily.
- ☐ I moved a lot but it worked fine
- ☐ I could not get anything working for me.

10. Did you find it easy to point on the projection screen with the "dot" and the infrared tower?

- ☐ Yes all of the times
- ☐ Most of the times
- ☐ No it was very difficult

11. Did you feel any strain on your neck or any other part of your body related to moving your head to point with?

- ☐ Yes on my neck
- ☐ Yes on my neck and shoulders
- ☐ Yes, on my shoulder
- ☐ No strain at all
- ☐ Other part of my body (please specify) ....

12. Comparing the two versions of the Presentation Conductor prototype that you used today, which is the order of preference for the particular interaction style you were asked to use? (1 favourite, 2 average, 3 did not like)

    ☐ Gaze and gesture (small screen)

    ☐ Head pointing and gesture (projection)

13. Additional comments on the technology

For each of the pairs below tick that you think is the most important for carrying out the task

1) Less Mental Demand ☐	Less Physical Demand ☐
2) Less Mental Demand ☐	Less Temporal Demand ☐
3) Less Mental Demand ☐	Less Effort ☐
4) Less Mental Demand ☐	Greater Performance ☐
5) Less Mental Demand ☐	Less Frustration ☐
6) Less Physical Demand ☐	Less Temporal Demand ☐
7) Less Physical Demand ☐	Less Effort ☐
8) Less Physical Demand ☐	Greater Performance ☐
9) Less Physical Demand ☐	Less Frustration ☐
10) Less Temporal Demand ☐	Less Effort ☐
11) Less Temporal Demand ☐	Greater Performance ☐
12) Less Temporal Demand ☐	Less Frustration ☐
13) Less Effort ☐	Greater Performance ☐
14) Less Effort ☐	Less Frustration ☐
15) Greater Performance ☐	Less Frustration ☐

TLX For PowerPoint Experiment

MENTAL DEMAND

Low High

PHYSICAL DEMAND

Low High

TEMPORAL DEMAND

Low High

PERFORMANCE

Good Poor

EFFORT

Low High

FRUSTRATION

Low High

TLX For Eye tracker Experiment

MENTAL DEMAND

Low High

PHYSICAL DEMAND

Low High

TEMPORAL DEMAND

Low High

PERFORMANCE

Good Poor

EFFORT

Low High

FRUSTRATION

Low High

TLX For Large Screen Experiment

MENTAL DEMAND

Low High

PHYSICAL DEMAND

Low High

TEMPORAL DEMAND

Low High

PERFORMANCE

Good Poor

EFFORT

Low High

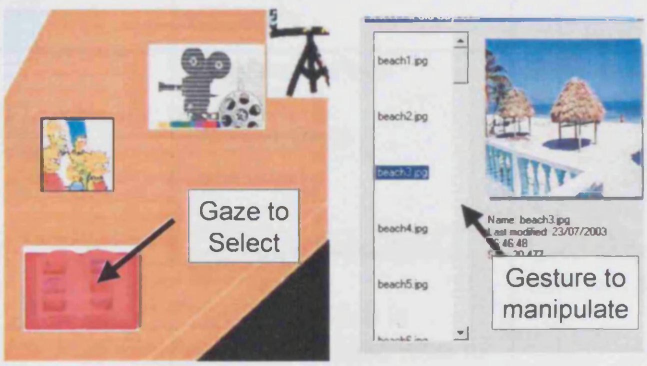
FRUSTRATION

Low High

Appendix F – User Guide

F.1 Two-phased interaction

The Presentation Conductor uses a two-phased interaction method. This is done by using each modality (gesture and gaze) in a specific way and in a particular function. The interaction between the user and the interface can be seen as a *dialogue* that is comprised of two phases. In the first phase, the user selects the on-screen object by *gazing* at it (for example, the user looks at the *picture book* object and activates it, as illustrated in figure F.1a). In the second phase, with a gesture the user is able to manipulate the selected object (for example, by pointing with the right hand the user can indicate that this is the gallery that he /she wishes to use) This combination of interaction modalities provides two-phased interaction.



(a) **(b)**

Figure F. 1 – How two-phased interaction has been implemented

F 2. Overview

The Presentation Conductor is a tool that supports the creation and display of multimedia presentations. These presentations can be comprised of a variety of multimedia objects that can be modified and arranged in the way that you will find suitable. The main features of the Presentation Conductor application allow you to:

- Create a presentation composed of multimedia objects
- Manipulate the multimedia objects
- Store created Multimedia presentations
- Display created presentations
- Achieve this using the Conductor Methodology

F 2.1 Incorporating the Conductor Methodology within the application

The Presentation Conductor utilises both gesture and gaze inputs. The gestures map to commands that invoke specific functionality. Table F.1 illustrates how each of these gestures is mapped to a specific functionality within the Presentation Conductor application.

Gesture	Name	Function
	***Left hand twist*** Hand performs a twisting act from right to left and vice versa	Used to adjust properties of a selected media object, by turning a knob.
	***Right hand twist out*** Hand in a fist performs a twisting action from left to right only	Used to play a presentation
	***Right hand Point*** No movement	Used to select an item from a list.
	***Left hand Point-Down*** Starting position is pointing up, hand twist to the right and ends pointing downwards	Used to delete a selected item and send t to the pit
	***Right hand Scroll Up & Down*** Hand performs a vertical flex movement	Used to scroll through the contents of the active Media Gallery.
	***Right hand Flex and hold Up & Down*** Hand performs a vertical flex movement, and pauses at the extreme high or low	Used to adjust vertical position or height of a displayable media object
	***Left hand "Stop"*** Hand raised flat in front of the user	Used to stop playback of a Media object or Presentation
	***Left hand vertical flat*** Hand performs a horizontal movement	Used to adjust horizontal position to the right of a displayable media object

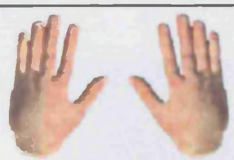
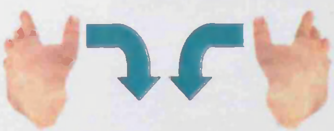
	***Right hand vertical flat*** Hand performs a horizontal movement	Used to adjust horizontal position to the left of a displayable media object
	***Hands Framing*** Both hands forming an index point, placed index to index	Used to initiate proportional size modification for displayable media objects
	***Both hands vertical flat, palms facing*** Both hands stretched, vertical to the ground, palms facing each other	Used to proportionally decrease the size of a displayable object
	***Both hands vertical flat, palms out*** Both hands stretched, vertical to the ground, palms facing out	Used to proportionally increase the size of a displayable object
	***Both hands flat*** Both hands flat, parallel to the ground	Used to finalize an operation.
	***Both hands "shut the box"*** Both hands stretched, vertical to the ground, palms facing each other, do an inwards movement, as if shutting a box	Used to Exit from the application

Table F. 1 Mapping between gestures and operations within the Presentation Conductor

The Presentation Conductor application translates a user's gaze into PC mouse movements. For example when you look at a Media Gallery on the stage, has the same effect as if you were moving the mouse onto that Media Gallery. Table F.2 illustrates the simulated mouse events that are handled by the Presentation Conductor.

Native Mouse Pointer Events	Description
MouseEnter	Used when the User looks at a selectable object
MouseLeave	Used when the User looks away from a selectable object
MouseMove	Used when the User looks at a new position on the screen

Table F. 2- How mouse events are mapped to functionality within the Presentation Conductor

The combination of these two interaction methods allows you to fully utilise the Presentation Conductor application.

The actual process of presentation creation itself is carried out in the environment that is based on the Orchestra metaphor. The Media Galleries on the stage represent the resources that can be used within the presentation, and the central stage is where the presentation is actually constructed (Figure F.2).

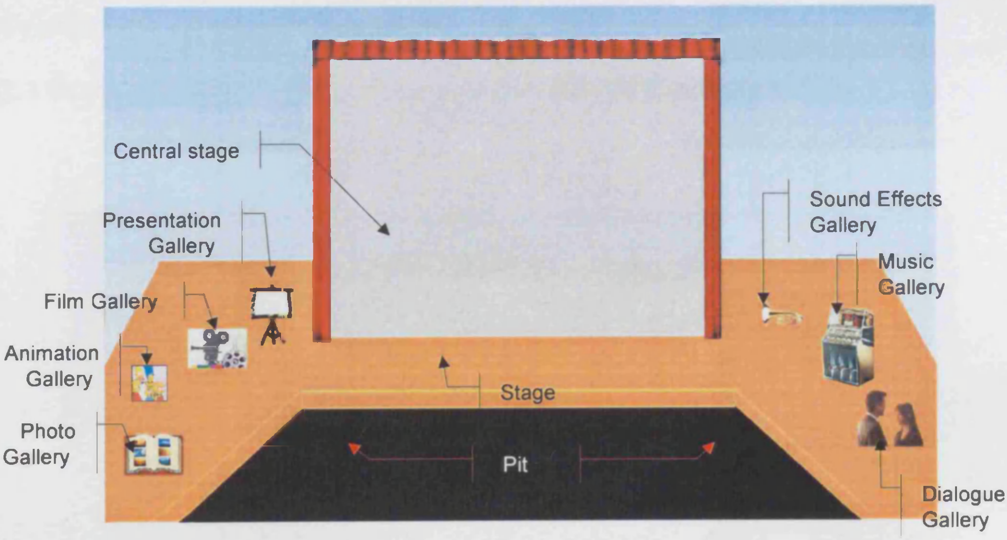


Figure F. 2 - The implementation of the Orchestra metaphor used with the Presentation Conductor

The range of "instruments", i.e. *media objects,* that are presented on the stage, are:

- *A Music Gallery (Jukebox)*: This represents various sound objects and includes music and environmental sounds.
- *A Sound Effects Gallery (Trumpet)*: This represents a series of sounds that the user can use.

- *A Dialogue Gallery (Two people talking):* This represents a series of short phrases that the user can use.
- *A Film Gallery (film camera):* This represents a series of films (i.e. digitised video sequences) that the user can use.
- *An Animation Gallery (Cartoon characters):* This represents a set of animations that the user can use.
- *A Photo Gallery (Picture book):* This represents a series of pictures that the user can use.
- *A Presentation Gallery (Projection Screen):* This represents previously created and stored presentations.

F 3. Supporting the creation and editing of presentations

The chief function of the Presentation Conductor application is to support users to create and edit multimedia presentations.

The actual creation and editing of a presentation can be broken down into three stages. Firstly, browsing and selecting the appropriate Media Object, secondly, editing its properties and finally, placing it within the presentation.

F 3.1 Supporting user browsing of the Media Galleries

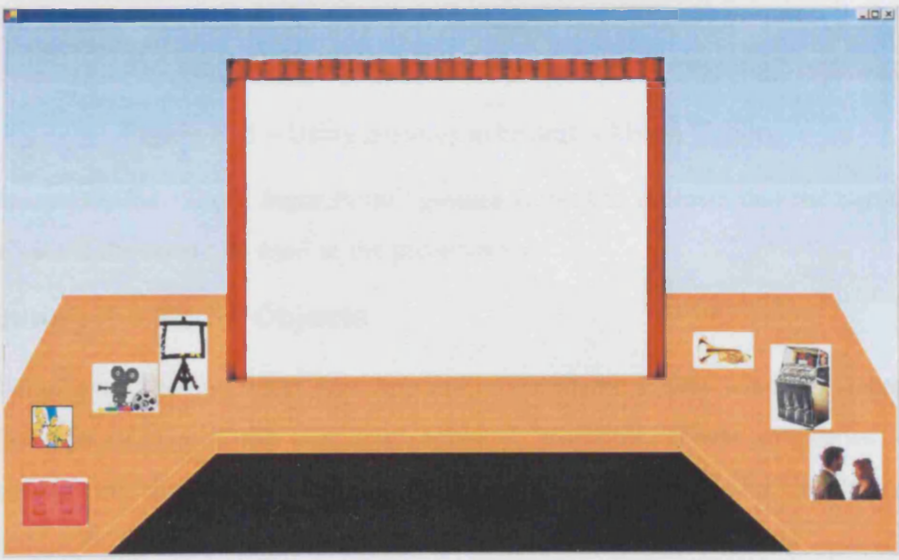


Figure F. 3 - Using gaze to select a Media Gallery

1. To select a Media Gallery the user has to look at the icon representing it on the stage. When the user focuses on a Media Gallery it appears highlighted. At that point the

user must do a "right hand point" to indicate that he/she wishes to open this gallery and access the media objects stored within it (as shown in figure F.3).

2. The Gallery gets activated and the Media Browser window is displayed. At this point all of the other galleries fade (as shown in figure F.4).
3. The Media Browser window displays in a scrollable list all the Media Objects stored in the gallery. To scroll through the list use the "***right hand scroll up***", and the "***right hand scroll down*** " gestures. Depending on the Media Gallery a preview of the media object will appear in the preview window (i.e. for an image, an animation or a video). For an audio-based object, the currently selected object is (as illustrated in figure F.4).

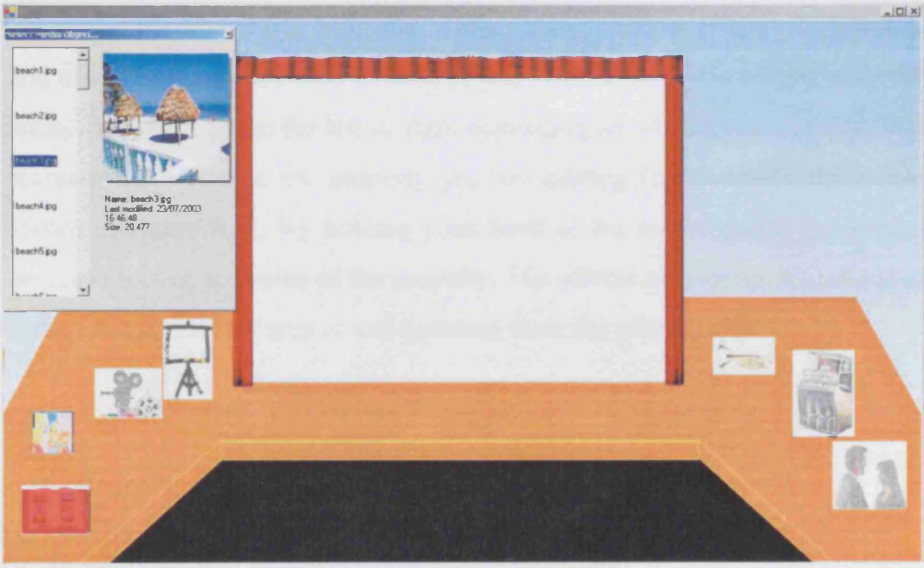


Figure F. 4 – Using gestures to browse a Media Gallery

4. Once again the "***Right hand Point***" gesture is used to indicate that the current media object is the one to be used in the presentation.

F 3.2 Editing the Media Objects

5. The selected Media Object now appears in the Media Editor window. Depending on the type of object, the user may adjust it properties. These properties and their controls are shown in Table F.3, together with the types of Media Object they can be used with.

Control Icon	Control Name	Control Description	Media Objects it Manipulates
	Brightness	Alters the brightness of an Image	Images from the Picture gallery
	Contrast	Alters the contrast of an Image	Images from the Picture gallery
	Volume	Alters the sound volume of the media object	Sound effects, Music. Dialogues, and Video
	Playback Speed	Alters the speed at which the media object is played back	Sound effects, Music, Dialogues, Animations and Video

Table F. 3 - The Media Controls that have been implemented in the Presentation Conductor

6. To select a Media control from the Media Editor window (Figure F.5) just look at it and then bring your left hand in front of you in a fist, as if you are grasping the control knob. Turn your fist to the left or right depending on whether you want to decrease or increase the value of the property you are editing (for example the brightness as shown in figure F.6). By holding your hand at the turned position you increase or decrease further the value of the property. The effects of your modifications are either shown in the preview area or can be heard from the playing clip.

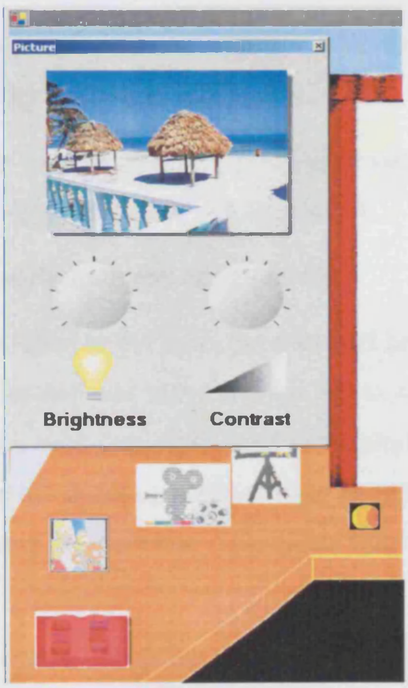


Figure F. 5 – The Media Editor

7. When you are finished with editing the specific property indicate this by putting both palms flat in front of you.

Figure F. 6 - Using the Media Editor to manipulate the properties of a Media Object

8. Repeat the same operation for applying the other control or reapplying the same control if you are not satisfied.

9. When you are finished with editing the properties, indicate this by putting once again both palms flat in front of you.

F 3.3 Placing, scheduling and deleting Media Objects

10. Your edited Media object should now appear on the central stage, or in the case of an audio object an icon representation will appear.

11. At this point you may want to adjust the position or size of the media object. The first step is to look at the object and point at it to select it.

12. You may resize or reposition in any order you like.

13. To move the Media Object to the right, put your left hand in front of you, in an open palm vertical to the ground and move it from left to right as if you are pushing the object. Do the opposite with your right hand in the similar posture to move the object to the right. To move the object up put your right hand flat in front of you and move it up to "pull" the object. In the same way move your hand downwards to push the object down.

14. To indicate that you want to resize the object form with both hands an index point and place them in front of you index-to-index. To make the object smaller place both hands in front of you with palms facing eachother and vertical to the floor and "squeeze" the object. The object will become proportionally smaller. To increase its

size, place your hands together in front of you, palms facing out, always vertical to the floor. Move your hands apart to "stretch" the object. The object size will increase proportionally.

15. To delete an object from the stage, select it as previously described and form a left hand point and flex your hand inwards as if you are directing the object to go to the pit.
16. To play a presentation, bring your right hand in front of you in a fist and turn outwards (as shown in Figure F.7).

Figure F. 7 - Playing a presentation

17. To stop the presentation playback at any time raise your left hand flat and vertical in front of you, in the "stop"/ "halt" position.
18. At any point that you want to stop a modification or selection you may indicate this by placing both palms flat in front of you.
19. To end the session, place once again both palms flat in front of you, and then vertically flat, palms facing in and move them as if you have a box with flaps that you want to shut.

Advanced features:

A presentation may have a number of scenes, each one containing a number of media objects. The first scene is automatically created when the Presentation conductor application is launched. In order to start a new scene form the "OK" sign your left hand and a new scene will be created, and its respective footlight will appear (as shown in Figure F.8).

Figure F. 8 – Placing a Media Object on the Central Stage

The schedule of the presentation is represented by a row of footlights that appear at the foot of the central stage (as illustrated in figure F.10). Each footlight represents a scene.

The Media Objects that are placed on the Central stage become part of the current scene. The current scene is indicated by the lit (yellow) footlight. The remaining scenes have their footlights grey.

To select a different scene to edit, look at the respective footlight and point with your right hand. You can now edit the objects on this scene.

One of the Media galleries is the Presentation Gallery, which stores the presentations that are created with the Presentation Conductor. If you want to access a presentation that you have previously created, either to play it or to edit it, you can select and browse the Presentation Gallery in the same way you would select and browse any other Media Gallery, as previously described. Once you have selected a presentation its scenes with their contents appear and you can navigate through them and edit the objects as previously described.

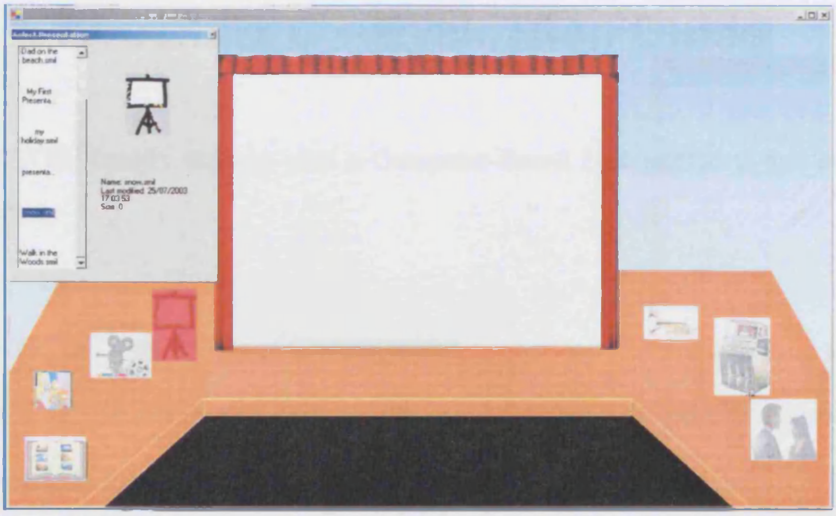


Figure F. 9 - Browsing and selecting a Presentation Definition

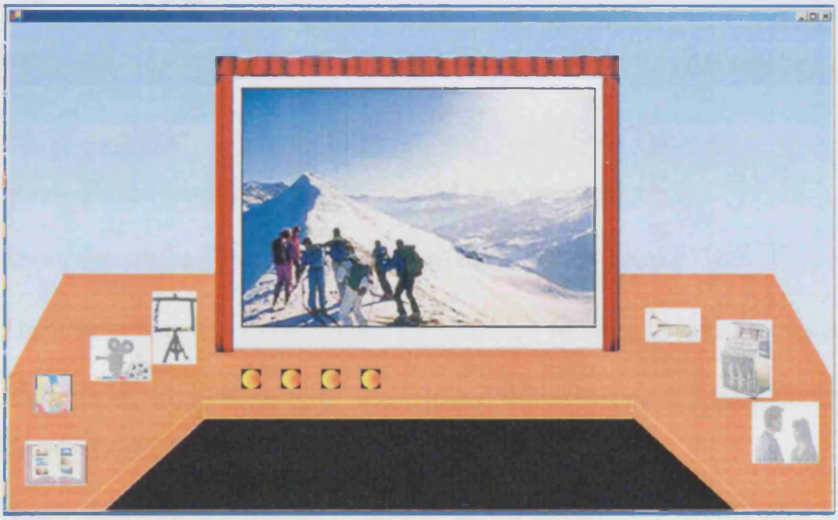


Figure F. 10 - A loaded presentation

Appendix G – Evaluation Guide

INVESTIGATOR: Briefly explain what a Computer-Based Presentation is and examples of their application.

Steps:

- Advantages over the use of transparencies
- The ability to incorporate media types that are available to computer users
- The ability to re use and edit previously created presentations.

Investigator: Give a brief overview of the system, and provide a quick walkthrough of it various features :

Steps:

- Show the subjects the Data Gloves and how they are worn
- Show the subjects the Eye tracker and how it is calibrated
- Show how to give a name for the presentation to be created in the session
- Show them the environment, and explain the use of each Media Gallery, and the Central Stage.
- Show how to select a Media Gallery, scroll through the Media objects, select an object, edit it and place it on the stage.
- Show how to add an object to the same scene and how to start a new scene
- Show how to play a presentation and how to resume editing
- Show how to exit the session

Investigator: Check to make sure that the user understands what has been said so far. Outline the user's role in the evaluation, that they will be asked questions throughout the process and that they can provide any comments at any time. Make the users aware that he whole evaluation process will be videoed with two cameras one showing their facial expressions.

Tasks

TASK 1: START A PRESENTATION

Q1: Do you find the instructions provide easy to follow?

Q2: Are you comfortable in your position at the moment?

Q3: Are the gloves fitting you hands well?

TASK 2: ADDING PICTURES TO THE PRESENTATION

Q4: How are you finding the use of the eye tracker (or IR tracker)? Can you locate your screen target easily?

Q5: Is it difficult to remember when to use gesture and when you select by gaze?

Q6: How do you find the adjustment of the brightness and the contrast?

TASK 3:ADDING A SOUND TO THE PRESENTATION

Q7:Do you have any difficulty in locating the Music gallery icon?

Q8: Was it easier to scroll through the gallery objects this time?

Q9: Was it easier to adjust the properties of the sound object, having the experience of adjusting the properties of the image?

TASK 4: ADDING AN ANIMATION

Q10: Do you have any difficulty in locating the Animation Gallery?

Q11: Was it easier to scroll through the gallery objects this time?

Q12: Was it easier to adjust the properties of the animation object, having the experience of adjusting the properties of the image and the sound?

Q13: Was it difficult to change the size of the animation (in the event that the user considered it necessary to do this)?

Q14: Was it difficult to move the animation to the new location?

TASK 5: PLAYING THE PRESENTATION

Q15: Did you find the Gesture used for this action appropriate?

TASK 6: EXIT THE APPLICATION

Q16: Did you find the Gesture used for this action appropriate?