McGill University

Redesigning the Automated Door Attendant: A Study of Multimodal Human-Computer Interaction for a Public Kiosk System

by

Michael Jonathan Perez
mp@cim.mcgill.ca
ID: 119831558

Supervisor: Dr. Jeremy R. Cooperstock

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Engineering.

Electrical and Computer Engineering Department
3480 University Street Room 633
Montreal, Quebec
Canada H3A 2A7

Montreal, Canada
October 2005

Your file Votre référence
ISBN: 978-0-494-22662-9
Our file Notre référence
ISBN: 978-0-494-22662-9

ABSTRACT

The Automated Door Attendant, a multimodal interactive system functioning as a virtual secretary for a professor, is an example of an artificial agent deployed in a role typically assigned to humans. Accepting both speech and touch as input, it serves as an appointment scheduler, a document browser, a videoconferencing client, and a video answering-machine for visitors. This thesis provides an analysis of existing multimodal kiosk systems, describes the evolution of the current system's design, and presents the results of an empirical user study.

RÉSUMÉ

Le Préposé de porte automatisé est un système d'interaction multimodale qui sert de secrétaire virtuelle pour un professeur. Celui-ci est un exemple d'un agent artificiel déployé dans un rôle typiquement assigné à un individu. Un utilisateur peut se servir de l'écran à touche pour naviguer l'interface graphique, cependant, le système est également capable d'interpréter des commandes par la voix. Le préposé sert de calendrier pour effectuer des rendez-vous, de navigateur de documents, de client de vidéoconférence et de répondeur-enregistreur vidéo. Cette thèse porte sur la recherche de solutions existantes, la conception et réalisation du système dévelopé et une étude effectuée suivi d'une analyse des résultats obtenus.

ACKNOWLEDGMENTS

Many individuals have contributed in a variety of ways to the production of this thesis over the last three years. First of all, I would like to thank my supervisor, Prof. Jeremy R. Cooperstock to whom I am most grateful. Jeremy took me on as a Masters student and has always been generous with his guidance, assistance, and patience throughout the development of this project. He continually provided me with valuable advice and priceless feedback, which will serve me well beyond the scope of this work. I'd like to express my deepest appreciation for Frank Rudzicz, who has been a most helpful friend to me. In addition, I'd like to commend Ali-Akber Saifee for his work on the speech recognition component of this project. I'd also like to thank Stephen Spackman for his ideas and clever suggestions, many of which greatly contributed to the design of the user interface. I am also grateful to Prof. Daniel J. Levitin, whose invaluable feedback has contributed to a number of improvements of this thesis. It would be impossible to ignore all those students who donated their time and participated in my research experiments for this project. In addition, I would also like to recognize the efforts of Steve DiPaola (Stanford University, CA) and his team for their work on Facade: Stanford Facial Animation System, which was used to model the ADA's 3-D avatar. Lastly, I must also thank all of the students whose work on the previous versions of the Automated Door Attendant has paved the way for this thesis project.

I would like to dedicate this thesis in loving and cherished memory of

Deniz Melek Sarikaya

1978 - 2005

Our dear colleague, friend, and angel.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

"Eventually, computers will take on the role of a good secretary."

- JAKOB NIELSEN [19]

1.1 Problem Description

Having one's own personal secretary is a luxury not often enjoyed in academic environments. Professors are often away from their office, making it difficult for students and other faculty members to get in touch with them. Other times, they may be talking to someone in their office, requesting not to be interrupted. Although telephone or email are often resorted to in these scenarios, it is natural for visitors to want to leave a quick message indicating that they stopped by. In other instances, a student may solely want to schedule an appointment, without disturbing the professor. While a pen-and-paper solution might appear to be the simplest to implement, the drawback is that professors may not see the note until the next time they return to their office. Such is the motivation for the creation of the Automated Door Attendant (ADA), a public kiosk mounted outside of a professor's office serving as a virtual secretary.

In continuous development since 1998, and undertaken by numerous students, it has become the subject of a complete rewrite and redesign for the purpose of this thesis. While preserving some of the core ideas of the previous version, the new design

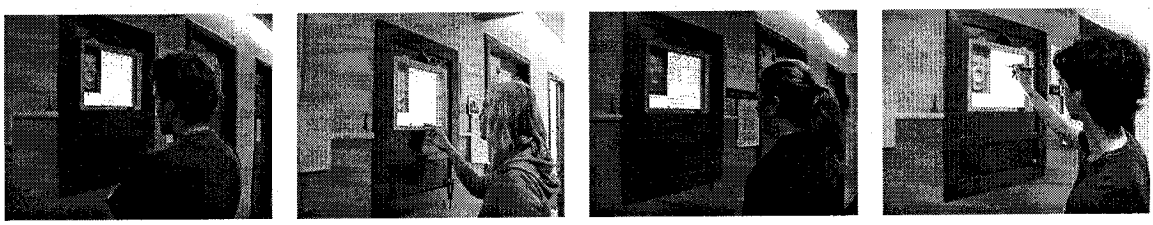

Figure 1.1: Users interacting with the Automated Door Attendant.

specifications require that multimodal interaction be possible using speech and touch, and that the interface implement a human-like conversational avatar. The rationale supporting these design objectives is to make conversing with the ADA as natural as possible, nearly simulating genuine dialogue with a human secretary. Additionally, it is required that the interface be extremely simple to use, such that a user does not have to waste time in order to accomplish a basic task.

1.2 System Overview

As shown in Figure 1.1, the kiosk is located on a wall outside of the professor's office. The only parts that are visible to the user are the mounted add-on touchscreen (with stylus) and monitor, however the system consists of a standard desktop computer connected to the network. A hidden pair of speakers and a microphone are plugged into the computer's sound card, and a small camera mounted above the monitor feeds into the video capture card. The microphone and speakers are required for speech I/O and the video camera is used for message recording, videoconferencing, and motion-detection.

The ADA greets the user with an image of a 3D-modeled human face and presents four choices (Fig. 1.2): viewing the professor's schedule, leaving a message, starting a videoconference session, or browsing documents. The anthropomorphization of the agent's appearance and voice are intended to encourage visitors to converse using natural speech. Being a multimodal system, the ADA can be navigated using any

Figure 1.2: The initial "home screen" that greets visitors.

combination of speech and touch commands.

If viewing the professor's schedule, the user may create appointments and leave written or video messages. The second menu item offers visitors the additional choice of recording messages without making an appointment. The videoconference option allows a student to initiate a live session with the professor participating from a remote location. Finally, the document viewer launches a web browser displaying a listing of viewable documents.

A separate web-accessible interface is available to the professor for performing administrative functions and accessing data from the ADA (see Fig. 1.3). The professor-side interface (PSI) allows one to view and edit the schedule for any date and view appointments along with any corresponding written notes. All of the logged events of the ADA are viewable using the PSI, as are all of the written and video messages left by users. Lastly, various system parameters of the ADA may be modified by navigating through the appropriate web pages of the PSI.

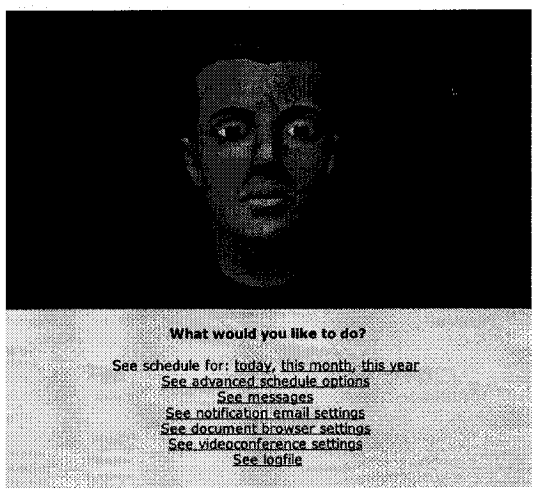

Figure 1.3: The initial web page of the Professor-Side Interface.

1.3 Contribution of Thesis

The main contributions of this thesis are the design and implementation of a kiosk system within a multimodal framework. The interface embeds an anthropomorphic conversational agent and is capable of accepting both speech and touch-based input from a user. Although initially designed to play the role of a virtual secretary for a professor, such a system could be extended and adapted to suit a multitude of applications.

Our contribution to the project consists of the design and implementation of all three software modules: the ADA front-end, the ADA speech module, and the professor-side interface. Preserving only basic functional requirements and applying lessons learned, the new system has been entirely redeveloped on a different platform, retaining none of the original software or code from past implementations. Except for the wooden kiosk enclosure and touch panel, the underlying hardware setup has been upgraded for the new system.[1]

The literature review places this work in the context of similar systems and iden-

[1]Details pertaining to the previous system are described in Appendix A.

tifies relevant points of discussion regarding multimodality and ECAs. The empirical study attempts to uncover the natural tendencies of users interacting multimodally and to derive meaningful data from the results obtained. Ultimately, this thesis serves to contribute to the body of knowledge in human-computer interaction, more specifically towards the advancement of research in multimodal interaction and the development of kiosk systems.

1.4 Layout of Thesis

The work presented in this document is organized as follows. The next chapter provides a survey of the literature describing existing relevant frameworks and technologies relating to kiosks and multimodality. Chapter 3 explores the development of the Automated Door Attendant system from its early stages through to its current implementation. An extensive analysis of the system's components is provided, with particular emphasis on the rationale for the approach and the design decisions made. In Chapter 4, a thorough evaluation plan is defined and the results obtained in laboratory experiments are presented. Lastly, some recommendations for future directions and concluding remarks are offered in the final chapter.

Chapter 2

Literature Review and Discussion

"The real problem with the interface is that it is an interface. Interfaces get in the way. I don't want to focus my energies on an interface. I want to focus on the job."

- DONALD NORMAN [21]

2.1 Multimodal Interaction

In the scope of human-computer interaction (HCI), there are minimally two participants: the human and the machine. Each are endowed with sensors and effectors for perception and control, respectively. Table 2.1 identifies those channels, or modalities, through which information is exchanged.

The taxonomy adopted to describe these different directions of information flow are as follows [17]: human-input channels (HIC), human-output channels (HOC), computer-input modalities (CIM), and computer-output media (COM). As outlined in Table 2.1, each of the human sensor-effector subsystems can be mapped to a corresponding computing device.[1] One of the aims of multimodal interaction is to obtain harmonious communication between human and machine, in which one's input channel can accommodate the information provided by the other's output modality.

[1]Only those communication channels relevant to the ADA are included.

Direction	Computer Modality (Device)	Human Modality (Organ)
COM - HIC	Monitor	Visual system (Eyes)
	Speakers	Auditory system (Ears)
HOC - CIM	Microphone	Articulatory system (Mouth)
	Camera	Motor system (Face/Body)
	Touchscreen	Motor system (Hand)

Table 2.1: Information flow in the context of the Automated Door Attendant.

According to Raisamo [25], "the feeling of using an interface can be faded out by not attaching any interaction devices to the user and by designing the dialogue in a way that really is natural and understandable for the user." The need for augmenting existing visual interfaces with additional modalities is easily justifiable when dealing with users who are visually impaired, however there are benefits of multimodal interfaces that are of universal interest to users. Although speech is often employed as an input modality in kiosks, users expect recognition to be reliable, otherwise they will reject it quickly. Preferably in this case, modalities should be redundant in that the user may execute a task in several ways. Complementary modalities may also help overcome weaknesses inherent to one modality by mutual compensation or by disambiguation of errors.

2.1.1 SpeechActs

Yankelovich's SpeechActs [30] is a telephony-based conversational speech system adaptable to a variety of applications. For the purpose of the author's study, it had been set up as an interface to an electronic mail and calendar application, allowing users to hear their messages and appointments remotely via telephone.

The system lets users call up and interact by speaking to the artificial operator

agent. They may request to listen to their emails, read back to them in a synthesized voice, or they may navigate their calendar and obtain event information. Essentially, SpeechActs serves as a front-end to existing desktop software, therefore voice commands are translated into actions within the back-end application. One of the challenges mentioned by the author is the conversion of an application's existing GUI into a speech-only conversational model. This involves redefining the manner in which information is organized and presented to the user, and modifying the language employed by the interface in order to adhere to conversational conventions.

The author describes the formative evaluation study in which fourteen participants[2] were asked to complete a set of tasks using SpeechActs. The experiment subjects performed their tasks from a telephone located in a quiet room and, along with some brief instructions, were given a reference card listing possible commands that the system could understand. The aim of the study was not to collect quantitative data, but rather to obtain users' feedback and to uncover usability problems.

Some of the issues that are immediately observed are recognition errors and ambiguous silence, both due to lack of appropriate feedback. It is a limitation of speech systems, when user input does not match the expected voice model, for commands to be rejected or misrecognized. The unpredictability of these errors distorts users' assumptions about cause and effect, and makes it difficult for users to formulate a conceptual model of how the system is supposed to work. Likewise, when silence is heard after a user utters a command, then it may either be due to the recognizer taking a while to process the input, or it may be due to the system not picking up any audible input. Regardless, the system should be better able to deal with these incidents by providing some sort of feedback that a command was heard and how it was recognized. An audible tone could cue users that their input has been heard and is being processed. Additionally, the system could issue a confirmation message or ask users to repeat their command if it was not recognized as a legitimate input. One

[2]No demographic details about the participants are available.

of the ways in which recognition errors can be reduced is by replacing open-ended questions with prompts that implicitly suggest that a limited range of responses is expected.

Following the study, Yankelovich describes some of the observed challenges inherent to the nature of speech. A speech-only user interface, for example, is demanding of users' memories and attention because of the fact that it requires a constant mental model of the current state. Furthermore, speech-only interfaces are limited by the amount of information that can be presented at each prompt, since speech is itself a slow output medium. These limitations, coupled with the lack of visual feedback, suggest that a speech-augmented GUI is preferable to a speech-only user interface (SUI) where the user has such an option available. We agree with this assertion because a multimodal system offers more interaction flexibility than a SUI, allowing users to explore the interface at leisure. Additionally, multimodal interfaces provide more opportunities for presenting feedback than in SUIs, especially during moments of ambiguous silence.

The author concludes by emphasizing the importance of designing speech-only and multimodal interfaces from scratch, rather than translating or evolving from existing graphical interfaces. For example, graphical interfaces typically contain terms that are unlikely to be used in casual dialogue, such as "delete", or "cancel", and generally lack relative concepts such as "next Monday" or "a week from tomorrow". Yankelovich's study demonstrates that most of these GUI conventions do not transfer well to speech interfaces, and that adhering to the principles of human conversation makes for a more usable interface.

2.1.2 Service Transaction System

The integration and synchronization of speech and pen input is explored in Oviatt's study of the Service Transaction System (STS) [22]. Essentially an interactive dy-

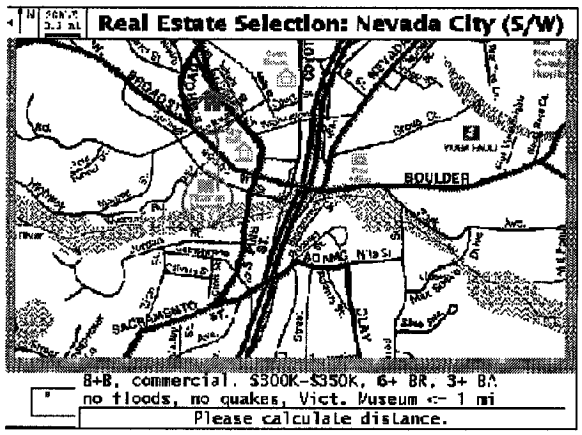

Figure 2.1: Screenshot of the STS.

namic map, this application allows users to speak commands while drawing on an LCD tablet. For example, a user may draw a line between two points while asking *"How far from here to here?"*.

The STS is currently tailored to be implemented in the real-estate domain, using the maps to locate homes and properties (see Figure 2.1). Although, both spoken and pen input modes can convey language, Oviatt claims that multimodal interaction during map-based tasks has numerous advantages, mainly because people have difficulty articulating spatial information.

A study of the STS was conducted to identify when users are most likely to interact multimodally rather than unimodally, and to investigate how spoken and written modes are synchronized and integrated. Eighteen native English speakers were selected to participate, with equal numbers of men and women, of varying ages and professions. They were assigned four tasks each, with order counterbalanced, resulting in data collected from a total of 72 completed tasks. Tasks included calculating distances on the map, labeling locations, querying information about points, and printing results. Prior to commencing experiments, subjects were requested to perform some "practice runs", completing entire tasks using only speech or only pen. Unfortunately, Oviatt does not provide any data from these practice tasks, therefore

there is no way of knowing how much subjects were influenced by this prior exposure to the system.

The study demonstrated that spoken and pen input modes were almost always used to convey complementary information, and it was extremely rare for information to be duplicated in both modes. Users expressed an overwhelmingly strong preference to interact multimodally when performing map-based tasks. The results indicate that pen input was used 100% of the time to convey location and spatial information, whereas speech was used for 100% of subject and verb constituents. It was also observed that writing usually preceded speech input, in that users elaborated their written marks by speaking about them. Oviatt hypothesizes that the act of drawing and the permanence of digital ink marks had an influence on users' subsequent speech.

Although the STS accepts user input multimodally, its output is restricted to the graphical display on the LCD tablet. We find that a system should exhibit a certain amount of reciprocity because it is unnatural to speak to something that is not responding in kind. It would be interesting to hypothesize whether the subjects would still have employed speech to this extent if they had not been asked to complete "practice runs" before using the system. Otherwise, the findings of the study represent an important contribution to the research in multimodal interaction.

As a result of the STS findings and subsequent experimental systems, Oviatt compiled the "Ten Myths of Multimodal Interaction" [23], an essay in which she challenges several common assumptions held by interface designers, and presents contrary empirical evidence for them. One important point made by the author is that a multimodal system is not necessarily more efficient than a unimodal one. Multimodality offers flexibility, allowing users to choose the modality that best suits their task, depending on the operating environment. Another somewhat obvious yet important claim is that some modalities can convey certain types of information much more efficiently than others, therefore it is delusive to assume that input modalities are completely

interchangeable. When possible, users will generally avoid employing error prone descriptions and will eliminate many linguistic complexities, if possible. However, when a recognition error occurs, users will likely try an alternate input modality rather than attempt again with the same modality. Oviatt's article cites many relevant issues to consider, and emphasizes the need for careful design of multimodal interface architectures.

2.2 Embodied Conversational Agents

Breaking from the traditional desktop metaphor, embodied conversational agents (ECAs) provide a collaborative dialogue between the user and a virtual attendant. Although often based on anthropomorphic models, they range from animated paperclips to complex human-like characters possessing their own "personalities". These assistants serve as proxies between the user and the system, shifting the focus away from the interface, in a figurative sense, and towards the task at hand. In a sense, commands are delegated to these agents rather than being executed directly by the user.

2.2.1 Agents with Faces

In an article entitled "Agents with Faces: The Effects of Personification of Agents" [12], Koda asks whether software agents should be personified and how realistic their appearance should be. Using a web-based multi-player poker game as an experiment platform, four different personified agents are depicted playing against each other and the user.

As shown in Figure 2.2, the four players are represented by a photorealistic human face, a human caricature face, a line-drawn smiley face, and a cartoon-like dog. Each agent is capable of displaying ten facial expressions, depending on whether they are

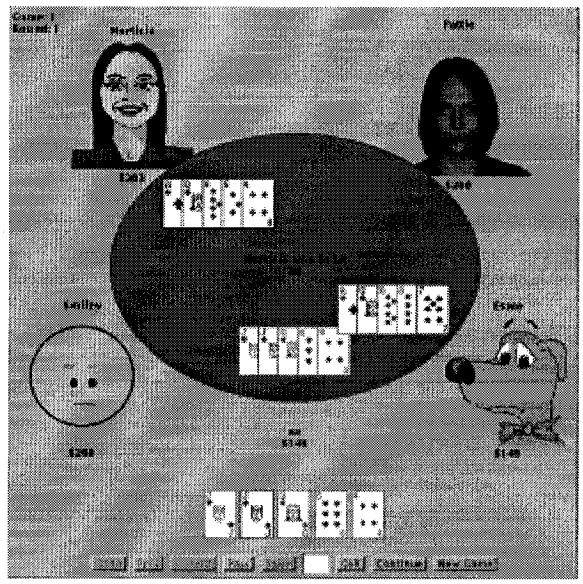

Figure 2.2: Four personified poker playing agents.

dealing, betting, bluffing, winning or losing.

The purpose of the experiment was to analyze the subjects' impressions of the agents during the poker game. Ten participants were recruited for the study, all of whom were graduate students at the MIT Media Laboratory, with the average age being 27 years. Although it is stated that their computer expertise level is advanced, nothing is known about their gender. Furthermore, the author provides no explanation as to why only advanced students from within the research laboratory were selected. We see this as a potential source of bias and conflict of interest, since the subjects may be familiar with the author and the existing literature.

Overall, users describe the experience as more engaging and likable than using non-personified interfaces, and reveal that they consider the player with the photo-realistic face to be more intelligent and comforting than the others. Koda's findings also show that users attribute human characteristics to ECAs and tend to respond emotionally to them. However, we find it peculiar that both of the human agents in the game are female, and that nothing is known about the gender of the experiment

Figure 2.3: REA welcoming a user.

subjects. It would be interesting to see the data presented as a function of gender to see if such an effect is present.

Finally, Koda warns that personifying an interface may lead to heightened expectations of the system's abilities. Careful consideration must be put into choosing an appropriate representation for an agent, and a compromise between perceived intelligence and friendliness should be obtained. Thus, ECAs with synthetic faces may often yield a more realistic estimation of a system's intelligence by users.

2.2.2 REA: Real Estate Agent

Cassell's real estate agent (REA) [3][4] is an embodied conversational agent capable of conversing with users, responding to their questions, and guiding them through pictures of rooms and properties. She generates and interprets speech and simple gestures, based on the state of the conversation. Citing the advantages of conversational interfaces, the author writes that "these communication protocols come for free in that users do not need to be trained in their use" since they exploit the natural affordances of the human body.

REA reacts to a user's presence and invites them to converse by raising her eyebrows and gazing at them (Fig. 2.3). She provides feedback that she is listening by

nodding or saying "uh huh" and "I see". In addition to recognizing human speech, she can interpret non-verbal cues such as gaze direction, body position, and a few simple gestures. The user, wearing a clip-on microphone, stands in front of a large projection screen, on which REA is displayed, while two cameras track the user's head and hand positions in space. We believe that wearing a clip-on microphone may be considered obtrusive by some users, therefore a well-positioned microphone located above the projection screen can improve this, assuming proper speaker placement.

The author does not describe any studies or evaluations of the interface, however it is mentioned that casual users of the system quickly entrain to REA and begin to nod and turn their heads in synchrony, within a couple of conversational turns. Although REA cannot yet entrain her non-verbal behaviors to those of her users, she possesses a wide repertoire of interactional output behaviors. By making subtle changes in eye contact, hand placement, and body orientation, REA can signal when she wants to speak and when she is ready to give the conversational turn back to the user.

Ultimately, the author stresses the importance of implementing conversational interfaces based on research in the social sciences relating to human-human interaction. The REA system demonstrates that it was designed with these principles in mind, and results of early trials look promising.

2.2.3 SmartKom

SmartKom [29], a multimodal dialogue system, merges speech, gestures, and facial expressions for input and output. Its interaction style is based on the situated delegation-oriented dialogue paradigm (SDDP), in which instructions are conveyed to an animated character known as Smartakus. The user and agent work cooperatively towards an intended goal, initially specified by the user. Smartakus can ask questions to obtain additional information from the user, and such a dialogue continues until

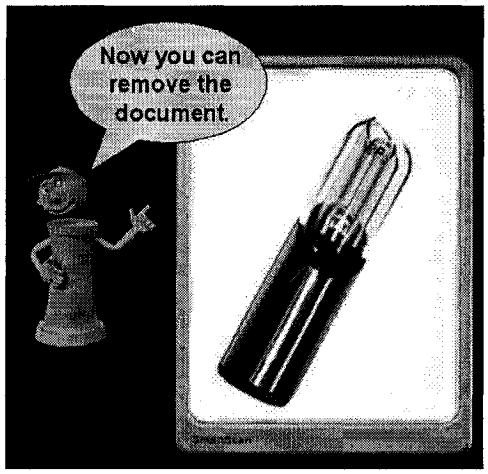

Figure 2.4: SmartKom's animated agent assisting a user.

the delegated task is executed by the system. Shown in Figure 2.4, the cartoonish agent has the shape of a lowercase "i", a symbol often associated with information stands.

The animated 3D assistant is capable of making pointing gestures and communicating various facial expressions to the user. In addition, it can also synchronize its lip movements to SmartKom's text-to-speech engine. Gesture recognition is achieved using an infrared camera pointed to the projection area of an LCD video projector, creating a sort of "virtual touchscreen". Allowable gestures can range from pointing with a finger to pushing a virtual button.

A user study of the system [28] gathered 35 subjects between the ages of 19 and 60 years. The subject pool consisted of 18 female and 17 male subjects, 24 of whom were students while 11 were employed. Participants were asked to use the system to purchase movie tickets and select seats using a cinema program designed for testing SmartKom. The speech portion of the study was implemented using a Wizard-of-Oz methodology, a paradigm in which a human experimenter simulates the role of the speech recognition unit. However, in this case, the speech output was also simulated by the experimenter.

Subjects were asked whether they perceived the SmartKom system behaved more like a person or like a machine. We find that responses to such a question are immaterial when Wizard-of-Oz testing is employed, especially in the case when speech output is also simulated. Regardless, those subjects who believed that SmartKom behaved more like a person (accounting for 30% of all subjects) actually spoke to the system politely, often saying "please", "thank you" or "sorry".

Although no meaningful information is provided by the authors regarding the system's abilities to interpret facial expressions, it is briefly mentioned that SmartKom is trained to "detect signs of annoyance" in users' faces. In the end, it is difficult to ascertain how well the system succeeds in interpreting users' queries, however we do find that this collaborative dialogue model offers a great deal of potential for kiosk applications.

2.3 Kiosk Systems

In the context of this work, a kiosk system refers to any enclosed machine aimed at self-service. This definition extends to devices such as banking ATMs, airport check-in machines, and tourist information systems. The kiosks explored in this section are restricted to those systems that accept speech input and are augmented with an embedded conversational agent.

2.3.1 Touch'n'Speak

Touch'n'Speak [16][24] is a multimodal information kiosk that can be configured for various applications. Users may interact with the system using the touchscreen or by speaking to the embedded interface agent in the corner of the screen (see Figure 2.5).

Figure 2.5: The Touch'n'Speak kiosk (left), and the avatar (right).

The kiosk is enclosed in an open-cabinet stand, and consists of a small camera mounted on a touchscreen monitor. The purpose of the camera is to allow the embedded avatar to follow the user's position and orient its head accordingly. The avatar is also capable of conveying a limited set of facial expressions while speaking via text-to-speech synthesis. Speech recognition is limited to command-based interaction in which only one- or two-word utterances are permitted. We feel that this severely limits the quality of the dialogue, because simply reading commands off the screen does not constitute conversational interaction.

For the purpose of testing the kiosk, a demonstration application for obtaining restaurant information was used. In total, 11 female and 12 male subjects were recruited, between 11 and 64 years old, and possessing varying computing skills. They were asked to repeat an assigned task three times, each trial using a different modality combination (in order): speech-only, touch-only, and both modalities.

The survey results indicate that users preferred multimodal to unimodal interaction. While speech was preferred over touch, test subjects would not have trusted speech as the only input modality due to recognition errors. Many users considered the interface agent to be annoying, especially when it behaved erroneously due to

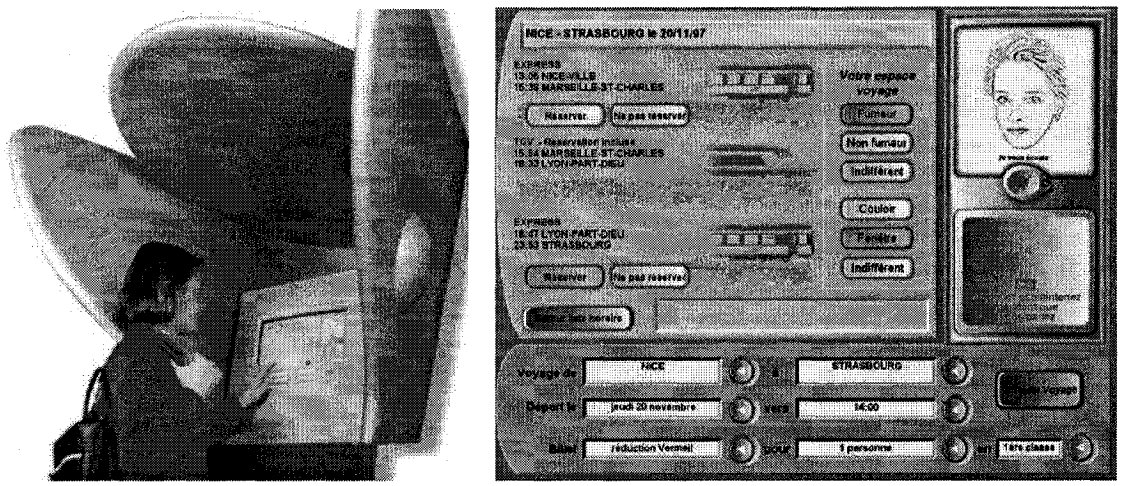

Figure 2.6: MASK kiosk (left) and GUI screenshot (right).

speech recognition or motion-detection errors. Moreover, several users reported not having noticed that the avatar displayed different facial expressions. These claims are not surprising when one considers the command-based interaction model being used, and the fact that the avatar occupies such a small area of the screen. For both of these reasons, we do not consider the interface to be very engaging and agree with the authors that a keyword-spotting speech recognition engine would help improve the quality of the dialogue with the user.

2.3.2 MASK: Multimodal Multimedia Service Kiosk

The Multimodal Multimedia Service Kiosk (MASK) [7][13] is a public kiosk deployed in a train station (see Figure 2.6), intended to replace the existing automatic ticket machines. Designed to accept speech and touch input, it serves as a ticketing agent, and also provides additional information about routes and schedules.

In the top-right corner of the touchscreen, an embedded agent guides users through the ticket-buying process and lets them know whether it is listening, speaking, thinking, or waiting for input. Spoken queries for ticket-selection are expected to be phrased such that the destination, day and time are specified in the utterance (e.g.

"I would like a round-trip ticket from Paris to Lyon today around noon."). Otherwise, the clerk agent prompts the user to fill in the missing details.

In order to deal with the possibility of users leaving the kiosk before completing their transactions, fixed time-outs are present throughout the system. When not in use, the kiosk displays an animated screensaver, illustrating the features of the system. Since there is no camera in the kiosk, approaching users cannot be detected, and must touch the screen to begin using the system. Being located in a noisy public area, MASK is equipped with side panels and a push-to-talk button in order to minimize the effect of background noise. In addition to providing some acoustic isolation, the side panels also serve to address the concerns of users who would be hesitant to speak to a kiosk in public. However, it was reported that 87% of test subjects would still use speech input if the kiosk were located in a busy train station.

User experiments were conducted in which 100 subjects were asked to complete a task in three different interaction modes: touch-only, speech-only, and combined. The authors mention that participants were randomly selected among passengers walking through the train station, so as to cover a wide range of ages for each gender (60 male and 40 female subjects were chosen). Unfortunately, it is not specified whether the order of the modalities was randomized across subjects in order to rule out learning effects.

In a post-experiment survey, the majority of users (80%) reported that they preferred using MASK to the existing automatic ticket machines, finding it fast and user-friendly. When asked to choose their preferred modality, there was not a significant difference between speech (53%) and touch (47%). The best performance was observed when subjects were allowed to mix modalities, since they were able to follow their preference and transaction times were consequently reduced. Forcing users to use their non-preferred modality led to longer transaction times, often taking 60% more time than in the preferred case. In conclusion, we find that the study demon-

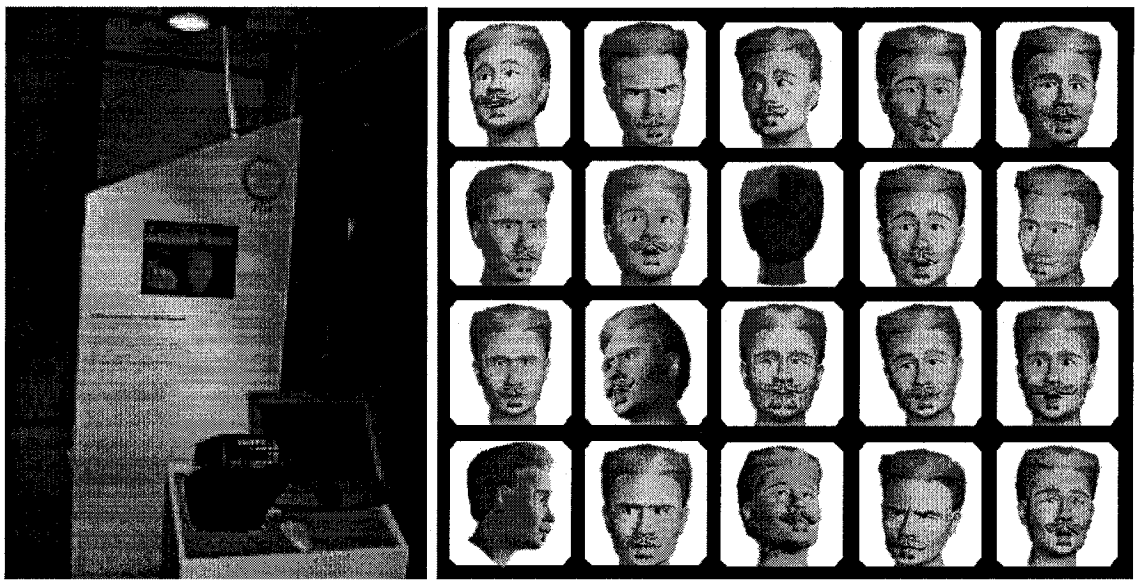

Figure 2.7: The August kiosk (left) and some of the agent's facial expressions (right).

strates that multimodal interaction is better suited than unimodal interaction for this application. Users' affinity towards each modality seems to be split between the two, and observed transaction times are lowest when modalities are mixed freely, since users can carry out their tasks by employing the modality that they perceive to be more efficient.

2.3.3 August Kiosk

Gustafson describes the August Spoken Dialogue System [9], a public kiosk featuring a sophisticated conversational agent. The 3D-animated head of the agent is capable of generating convincing lip-synchronized speech and exhibiting a variety of non-verbal expressions, as shown in Figure 2.7. Users communicate with the system solely by means of voice input, and it responds with synthesized speech or by displaying the requested information on the screen.

The kiosk features two screens (Fig. 2.7), although neither of these responds to touch input. The August animated agent is displayed on the rear monitor, whereas

the frontmost display is used for presenting textual and graphical information to the user. A (click-to-talk) mouse is located below the screens, and one of its buttons must be held down while commands are being spoken to the system. An advanced speech recognizer in conjunction with a dialogue manager ensures that August generates intelligent responses to users' queries. The kiosk also makes use of a hidden camera for detecting movement and for following the user's position so that the agent's head and eyes adjust accordingly.

Over the course of six months, the kiosk was deployed in a cultural center without supervision and used casually by over 2600 people (out of which 50% were men, 26% women, and 24% children). The aim of the study [10] was to collect speech data from random visitors in order to analyze how they interact with a spoken dialogue system. For the purpose of testing the system, August was configured to present information about restaurants and other facilities in Stockholm. Upon transcription of the logs and recordings, the authors measured that 60% of words used by visitors were related to information seeking, whereas the remaining 40% of words were uttered in a socializing context. It was also observed that users often modified their pronunciation and articulated their words with longer inserted pauses, especially when repeating misunderstood commands. Although the authors do not state how well the system succeeds in recognizing users' speech, it is mentioned that attempts to re-utter misunderstood commands accounted for 16% of all recorded words.

In terms of usability, the kiosk could be improved by eliminating the click-to-talk mouse and replacing it with a simple button. Additionally, the distance between the two screens should be minimized to avoid having users glance back and forth between the displays. Otherwise, we find that the unsupervised real-world study is a useful method of observing users behaving naturally and spontaneously, thus providing the experimenters with valuable data collected from users of the intended demographic.

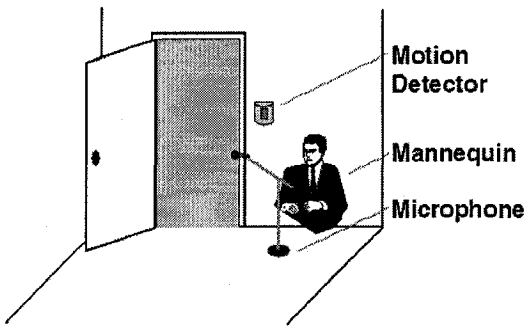

Figure 2.8: Setup diagram of the Office Monitor.

2.3.4 Office Monitor

Yankelovich's Office Monitor [31] is a physically embodied kiosk implemented as a lifelike mannequin. It is described by the author as the office equivalent of a telephone answering machine, allowing the office occupant to record a greeting before leaving so that passing visitors may listen to it and leave messages.

Dialogue is initiated by the Office Monitor, upon detecting motion, and is carried out using the microphone and speakers, as illustrated in the setup diagram (Fig. 2.8) provided by the author. Built using the SpeechActs [30] framework, the underlying speech application is very brief and lightweight, and is limited to recording audio messages along with a name.

The author initially conducted a short and informal pre-design study, in which brief encounters with a human secretary were recorded. These conversations were observed to be short, ranging from 10 to 20 seconds and exhibiting a common greeting-question-answer pattern. The formal study of the Office Monitor prototype involved 20 office colleagues who were asked to interact with the system, without receiving any set instructions or tasks. The interaction times with the system were considerably longer than in the pre-design study, ranging from 20 seconds to 2 minutes. However, 61% of participants rated their experience as "overall positive" and 28% as "mixed".

It is immediately apparent that this system is quite similar in purpose to the ADA, albeit lacking in terms of overall features and visual feedback. We find that the pre-design study serves as a proper basis upon which to model the Office Monitor's dialogue, and it serves as a useful benchmark comparison to evaluate the system's performance. On the other hand, we feel that the entire setup, including the mannequin, seems to occupy an excessive amount of space and would be obtrusive in most office settings.

In conclusion, Yankelovich suggests that the system would be improved if reconstituted as "an intercom device that attaches to the outside wall of an office [...] with an animated character to serve as a conversational partner". The author notes that such a system should employ a multimodal approach, combining speech with graphics and video, in order to more effectively address users' needs.

2.4 Discussion

As Donald Norman attests [21], traditional interfaces "get in the way" of productivity. They require too much effort to figure out, and they are generally not designed from a user-centric approach. The systems reviewed in this chapter present different forms of human-computer interaction aimed at improving the user experience. Since kiosks are likely to be located in public spaces, they will undoubtedly be used by individuals possessing little or no familiarity with computers. Multimodal input and embodied conversational agents offer the potential of rendering kiosks accessible to a greater user base, since they only require elemental communication skills to operate.

Almost all of the systems described in the chapter utilize speech recognition technology, often in combination with another input modality such as touch/pen or gestures. Depending on the application, Oviatt [23] mentions that some modalities are better suited than others for completing a given task. Pen input was demonstrated to

be useful for handwriting or map-based tasks [22], whereas gesture input is shown to be appropriate for pointing or selecting [29]. Given the basic functional requirements of the Automated Door Attendant, we judge that a touchscreen coupled with speech input is a logical choice for its implementation. A user could navigate the interface and make selections using speech or touch, and a stylus could be used for leaving handwritten notes on the screen.

It is important, as Yankelovich argues [30], for multimodal interfaces to be designed from scratch rather than treating speech input as an afterthought of GUI design. As a counterexample to this principle, the speech recognition component of Touch'n'Speak [16] appears to have been implemented as an extension to a touchscreen system, evidenced by the fact that speakable commands are limited to button text. Consequently, our ADA interface has been entirely redesigned as a conversational system, ensuring that it can be navigated either with the touchscreen or via natural language dialogue. The language employed throughout our system is intended to resemble casual human speech, opting for commands such as *"(I'd like to) Do something else"* instead of *"Cancel"* (although both are interpreted correctly).

Background noise and feedback are common problems associated with speech recognition implementations, although several solutions help minimize their effect. Whereas the REA system requires a wearable clip-on microphone [3], some of the systems have opted for a push-to-talk solution [7][9][13]. The affordances of a push-to-talk button provide the user with instantaneous feedback that the system is listening, while also simplifying the barge-in issue.[3] The obvious disadvantage, however, is that users must always have one of their hands occupied while speaking. The least obstructive solution would involve using a unidirectional noise-canceling microphone positioned at an appropriate height, preferably hidden from view.

In terms of vision, some kiosks employ a basic motion detector [31] whereas oth-

[3]Barge-in refers to the interruption caused when users want to speak while the system is speaking.

ers make use of a camera [3][9][16][29]. In addition to detecting movement, the video stream from the camera may be used for tracking and recognition purposes, something especially beneficial in immersive environments such as Cassell's REA. For now, the vision capabilities of the ADA are limited to detecting approaching users, however head-tracking will be implemented in the near future.

Returning briefly to Koda's study of "Agents with Faces" [12], and comparing the different ECAs evaluated in the previous sections, we believe that a synthetic human face and voice offer an appropriate representation of the ADA's conversational abilities and help achieve a rich level of interaction. While sophisticated agents such as REA, Smartakus and August are impressive, we do not feel that the ADA would benefit immediately from such an advanced design. Interaction time with our kiosk is quite short, around one minute, and the current level of task complexity does not require an emotionally expressive avatar. It is clear, however, that a minimum level of entrainment is necessary, therefore the ADA agent occupies a significant area of the screen, as opposed to systems such as MASK and Touch'n'Speak whose smaller avatars are restricted to the screen corner. Eventually, the agent should be augmented with the ability to follow the user's position using a head-tracking algorithm.

A variety of interface evaluation methodologies have been explored, including controlled empirical experiments, Wizard-of-Oz testing, and long-term unsupervised studies. While unsupervised studies offer the possibility to observe numerous unsuspecting users interacting with the system in its intended environment, it is then difficult to compare subject data equivalently due to changing conditions and variables. We feel that a controlled study would allow us to closely monitor the experimental conditions and the selection of subjects. Several studies employed a repeated measures design with isolated unimodal and combined multimodal interaction [3][13][16][22]. Ultimately, as we describe in Chapter 4, such an experiment format with counterbalanced orders allows us to measure and compare users' relative performance among the different modalities with a practical number of subjects.

Chapter 3

Design and Development

"Once the product's task is known, design the interface first; then implement to the interface design."

- JEF RASKIN [26]

3.1 System Architecture

The Automated Door Attendant kiosk maintains roughly the same hardware setup as its earlier implementations. Essentially, the kiosk machine is a standard desktop PC located inside the professor's office. In the corridor outside the office, a resistive analog touch panel overlay is mounted in front of an LCD monitor (see Figure 3.1). A CCD videocamera, connected to a frame grabber in the PC, is positioned above the monitor and aimed towards the user. Finally, a pair of speakers and a microphone are hidden behind a grille below the monitor.

The software system we have developed is composed of three independent modules: the ADA front-end, the ADA Speech Module (ASM), and the Professor-Side Interface (PSI). They are independent to the extent that each can run on its own, and their modular design ensures that each can be replaced without affecting the performance of the other components. The ADA is the sole interface with which the user interacts, however, the added benefit of the ASM running in the background

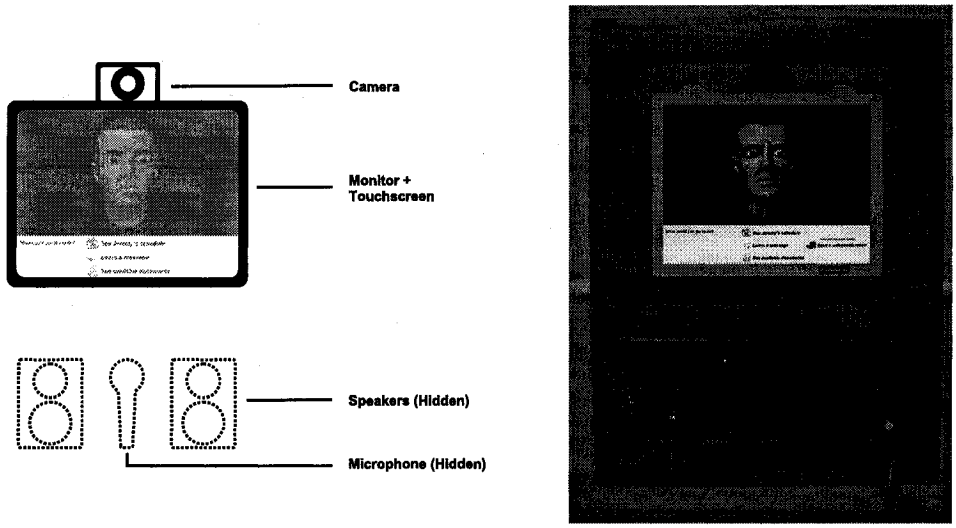

Figure 3.1: Architectural view (left) and photograph (right) of the hardware.

allows the user to navigate the kiosk using voice commands. The PSI is the back-end system that controls most of the functionality and default settings of the ADA from a web server.

As illustrated in Figure 3.2, synchronous communication between modules only occurs with the ADA and the ASM. The ADA sends state information to the ASM, and the ASM provides textual transcriptions of the recognized voice commands to the ADA. The PSI and ADA share data asynchronously through the use of text files, and changes generated from within either system are applied immediately.

While the ADA front-end makes calls to numerous external programs in order to function, these are not considered part of the system developed within the framework of this thesis.

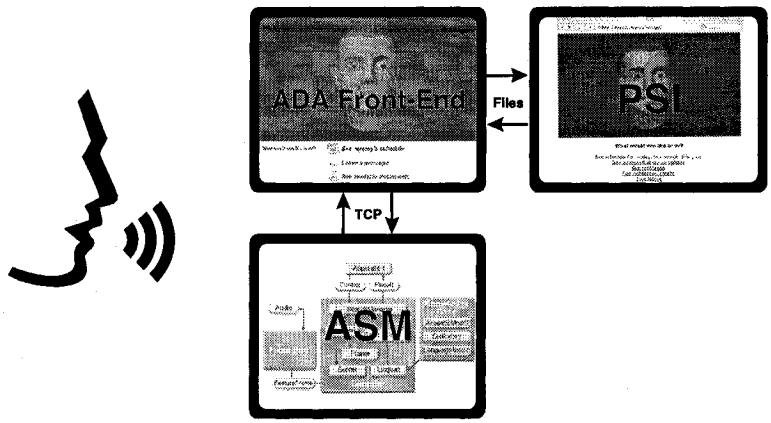

Figure 3.2: Software system architecture diagram.

3.2 ADA Front-End

The front-end of the Automated Door Attendant consists of the software responsible for the operation of the kiosk. In addition to communicating with the ASM, it launches processes and executes system calls for many of its features. Its touchscreen interface presents context sensitive buttons that change over the course of interaction.

3.2.1 Functional Overview

Upon being presented with the main menu (shown earlier in Figure 1.2), a user may choose to: *"See Jeremy's schedule"*, *"Leave a message"*, *"See available documents"*, or *"Start a videoconference"*. The five screenshots in Figure 3.3 show the system in various states, depending on the selection desired.[1]

When requesting to see Jeremy's (the professor's) schedule, the timetable for the current workweek is displayed on the screen, highlighting the current day and identifying the available appointment slots (refer to Figure 3.3a). The user can then request to see the following week's schedule or ask the agent to suggest some available times.

[1]Refer to table B.1 in the appendix for the complete state transition table.

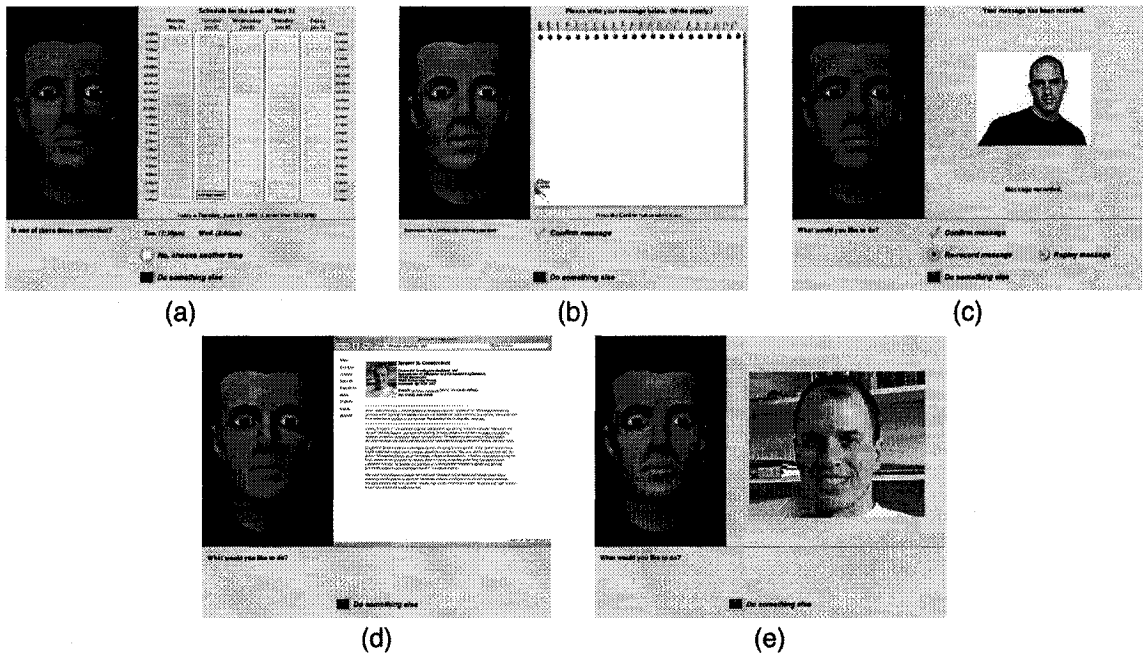

Figure 3.3: Screenshots of the ADA front-end in different states.

Alternatively, the user can simply speak (or press) the desired time-slot to book an appointment with the professor. An image of a paper notepad is presented prompting visitors to write their name with the stylus and confirm the appointment. If they prefer to record a video message instead of simply writing their name, then this option is available as well.

The message option from the main menu leads the user to a screen offering to choose between leaving a written note (Figure 3.3b) or a video message (Figure 3.3c), without making an appointment. As in the scheduling option, video messages can be recorded, replayed and rerecorded as often as desired before confirming the message. It should be noted that notification emails are sent to the professor whenever an appointment or message is confirmed with the system. The email message includes a clickable URL to view the written note (PNG file) or video message (AVI file) from the PSI.

Upon opening the document browser (Figure 3.3d), the Mozilla[2] web browser is launched with a list of viewable documents made available by the professor.[3] From there, the user may navigate the web page with the touchscreen or return to the main menu.

The fourth option in the main menu is only enabled when the professor is reachable via videoconference, otherwise it remains hidden from the user. When available, it initiates a videoconferencing session and displays a live video window on the screen, allowing the user to terminate the session at any time (Figure 3.3e).

3.2.2 Interface Design

Maguire [15] explores a number of design guidelines relevant to kiosk systems. Firstly, a kiosk must be noticed by passers-by and its purpose must be clear, since users will most likely be using the system for the first time. Such a system is to be used on a casual "walk up and use" basis, and should automatically return to its initial state when abandoned. The author warns that interaction time with a vertical touchscreen kiosk should be kept to a minimum, since prolonged use can cause arm fatigue.

The ADA kiosk is immediately awoken from its screensaver state when motion is detected in front of the camera. In addition, timeout counters have been strategically inserted throughout the system in order to deal with user inactivity. Consquently, the system will automatically slip back into screensaver mode when inactive for one minute (or longer, depending on the state). This notion of a system reset serves as a courtesy for the next visitor to use the system from its intended starting point.

Through an iterative user-centered design process, the redesigned interface has

[2]http://www.mozilla.org/

[3]Currently, the document browser displays a provisional web page, due to the fact that the complete list of documents has not yet been put together.

evolved from a paper storyboard into a functional prototype. By adhering to Jakob Nielsen's [19] formative evaluation guidelines, the interface is changed and retested whenever a usability flaw is noted. Design principles [20] such as consistency, visibility and feedback are enforced throughout the system.

The screen configuration remains essentially the same throughout the different states: The top three-quarters of the screen are allocated for displaying the agent's face and the main working area, whereas the navigation pane at the bottom is reserved for context-sensitive menu selections and additional feedback. Common amongst almost every state of the system, a menu item labeled *"Do something else"* is available to users for canceling the current action and returning to the main menu. Alternatively, shortcut commands such as *"Cancel"*, *"Go back"*, or *"Home"* can be spoken to obtain the same result.

To minimize the user's cognitive load, a maximum of four menu items are present in the navigation pane at any given state. Menu buttons, amply large enough to be pressed by a typical finger on the touchscreen, are accompanied by easily recognizable graphical icons and exhibit a brief 3D pressed-in effect when touched (or spoken). In addition to the instantaneous feedback provided upon selecting a menu item, the desired outcome is generally produced within a fraction of a second (or varying between 0.5 and 2 seconds when the command is issued using speech). In keeping with the responsiveness of the system, modality inputs are integrated sequentially such that the system responds to the command it receives first.

The state of the system is always visible to the user with the appropriate and consistent use of colors and textual labels. **Boldface text** represents instructions and/or questions, whereas *italicized text* represents commands that the user can speak or touch.[4] Hence, any clickable menu item can alternatively be selected by

[4]Although it is not known whether users notice or learn these font-style variations, it was observed during the empirical study (Chapter 4) that they had no difficulty discerning instructions/questions

speaking its assigned textual label. While in video record mode, for example, a large red countdown timer is shown on the screen alongside the video feedback window. In the schedule, available time slots are labeled "AVAILABLE" and are highlighted in blue. The color green, as in most interfaces, is associated with video playback and confirmation of messages.

When a user speaks a command, the bottom-left corner of the navigation pane displays the phrase as it was recognized. This element of feedback assists the user in determining which part of the request was misunderstood in the event of a recognition error. Another instance where feedback is required is at startup, upon when the ADA detects and greets the user by triggering the main menu and making a slightly audible throat-clearing sound.[5]

3.2.3 Software Development

The ADA front-end, developed in a Linux environment, is written in C using the GTK+[6] libraries for the windowing and user-interface elements. Within the GTK+ framework, Pango[7] is required for laying out and rendering all of the text throughout the system, particularly for the schedule grid. Although such a kiosk application could have been implemented using a multimedia authoring software, this would not have been favorable due to the fact that advanced programming and functionality is required. For example, the ADA requires the ability to launch and kill processes and embed external windows.[8] Additionally, OpenGL support is necessary for the ability

from actions/commands.

[5]The "throat-clearing" greeting was chosen for its brevity as well as for its subtle manner of implying "Yes. May I help you?".

[6]http://www.gtk.org/

[7]http://www.pango.org/

[8]Due to limitations of the GTK+ implementation used, the window manager was customized to "decorate" windows appropriately without borders rather than embedding them into the application.

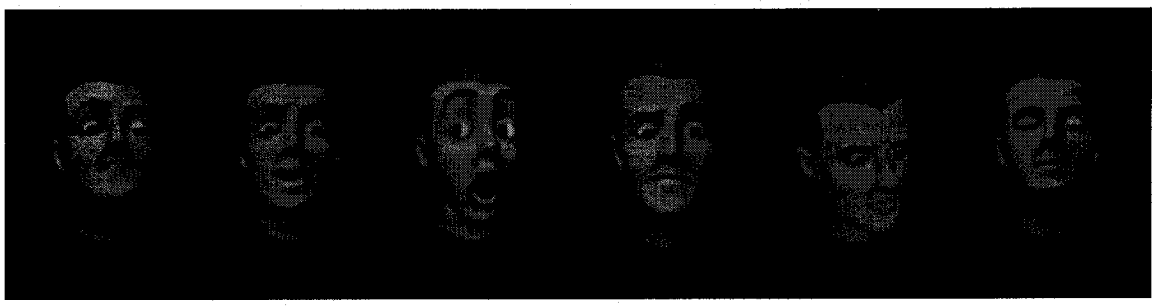

Figure 3.4: Various facial expressions generated with the Facade program.

to embed the 3D face of the agent.

File handling is an issue that required careful planning so that user files and messages can be remotely accessible via the professor-side system. All system parameters, schedule data, and visitor messages are stored in a common folder shared with the PSI. A global preferences file is also used to keep track of the paths of all images, external programs and scripts used by the system, such that they can be located prior to launching the system.

3.2.4 Anthropomorphic Agent

The ECA of the Automated Door Attendant, although presently limited to a static image of a face, is currently progressing into a more evolved agent. The avatar is displayed using an OpenGL program called Facade [6], which uses a fixed polygon face topology to define a list of several hundred parameters corresponding to facial features. These feature parameters can be dynamically changed to different values in order to generate a variety of character types or expressions, as depicted in Figure 3.4.

At this point in development, the agent does not employ any artificial intelligence algorithms. Phrases spoken by the system consist of prerecorded prompts, therefore it is still too early to describe it as being truly conversational. The most recent version of the Facade software offers the ability to lip-sync the agent's mouth with a

text-to-speech engine, therefore the potential to augment the system exists. Efforts already in progress, using basic image processing techniques, will allow the agent's eyes to follow the user's position.

Mentioned earlier, was the idea that using a photorealistic human face might mislead users into overestimating the system's intelligence [12]. Therefore adopting a 3D synthesized anthropomorphic face, as seen in other systems [24][9], projects a more realistic image of the ADA's current abilities.

3.2.5 Video Capture and Transmission

Video plays an important role in the functioning of the system, captured with the CCD camera through the frame grabber in the system. While the ADA is in screen-saver mode, the motion-detection subsystem is responsible for grabbing successive frames at one-second intervals and evaluates a function to measure the difference between them. When the difference is greater than a specified threshold, the ADA identifies this change as motion, and greets the user: the screen-saver exits, the screen displays the main menu, and the agent makes its throat-clearing "Hmm." sound. While this basic motion-detect algorithm may be improved to deal with high-traffic environments, it is sufficiently suitable for the location in which the ADA machine is currently set-up.

The video message recording feature requires *xawtv*[9], a video capture and viewing software utilizing the v4l API of the Linux kernel. The ADA calls a script that sets up the recording parameters and begins streaming audio and video into a MJPEG-compressed AVI file. The user can watch the video on-screen as it is being recorded, and then has the option of playing back the recorded file, handled using *mplayer*.[10]

[9]http://linux.bytesex.org/xawtv/

[10]http://www.mplayerhq.hu/

Videoconferencing is implemented using the McGill Ultra-Videoconferencing (UV) system.[11] It allows sending and receiving of low-latency audio and video at high resolutions in order to provide face-to-face interaction between the visitor at the ADA kiosk and the professor at the remote end.

3.3 ASM: ADA Speech Module

The ADA Speech Module (ASM) originated from a proof-of-concept project [27] utilizing the CMU Sphinx speech recognition system. We, then, rewrote it into a single independent back-end module capable of two-way interaction with the ADA front-end using TCP socket communication.

3.3.1 Recognition Engine

After exploring numerous speaker-independent speech recognition solutions for Linux such as the SpeechWorks OpenSpeech Recognizer and IBM's ViaVoice, it was determined that the CMU Sphinx-4 Speech Recognition System was the most compatible and affordable option. The Sphinx project is an ongoing effort currently in development at Carnegie Mellon University, in conjunction with Sun Microsystems Laboratories, Mitsubishi Electric Research Lab, and Hewlett-Packard's Cambridge Research Laboratory.

One of the most attractive aspects of the Sphinx system is its customizable architecture, shown in Figure 3.5. The FrontEnd component of the Recognizer has been rewritten into a simple command-line message-passing interface. Coded in Java, the modified FrontEnd consists of a thread that waits for a voice command and then

[11] http://ultravideo.mcgill.ca/

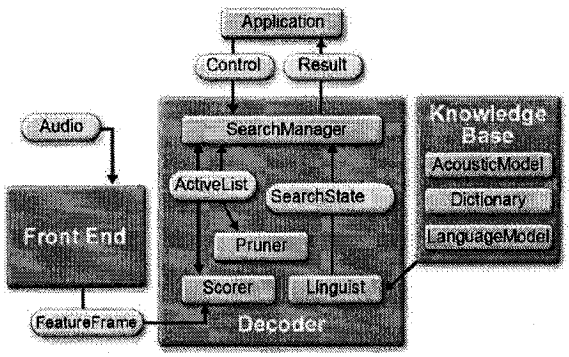

Figure 3.5: The Sphinx-4 system architecture.

passes it to the decoder and linguist. The LanguageModel grammar, described in the following subsection, restricts the search space of allowable words for the context of this application. The result is a parsed textual representation of the recognized phrase spoken by the user, such as *"I'd like to leave a written message please,"* which is then passed to the ADA for interpretation. The thread in the FrontEnd also monitors the TCP ports for incoming messages from the ADA, communicating the system's state. For example, the Recognizer is disabled in states during which the user is recording a message or participating in a videoconference.[12]

For the purpose of this application, the speech recognition engine is not expected to produce an exact transcription of voice commands issued by the user. Requiring this would necessitate an impossibly large and complex LanguageModel grammar definition, severely hindering system performance. Instead, a sufficient language model can be devised in order to accommodate the most common expected user input. Ultimately, the ADA parses messages upon receiving them from the ASM and searches for keywords in order to decide what selection the user intended to make, thus discarding extraneous words.

[12]Specifically, the states during which speech recognition is disabled are: 7, 8, 14, 15, and 20 (Refer to Table B.1).

3.3.2 Grammar Definition

The CMU Sphinx platform employs the Java Speech Grammar Format (JSGF) for textually representing the global grammar. The convention adopted by this format for structured syntax description is similar to Backus Naur Form (BNF). This notation allows one to define complex rules using simple tokens, logical operators, and weights.

Consider the following example rule:

```
<mycommand> = [ i'd like to | i want to ] (/2/see | /1/view) [the] schedule;
```

This would instruct the recognizer to accept several combinations of phrases such as *"View schedule"* or *"I'd like to see the schedule"*. The vertical bar serves as a logical OR, implying that one of the alternatives may be spoken. The square brackets surround optional parameters, whereas the round brackets serve to group expected tokens. The numbers surrounded by forward slashes are weights, corresponding to the likelihood of the alternative. The above example assumes that the verb *see* is twice as likely to be spoken as its alternative, *view*. While choosing an appropriate value for a weight in advance is difficult, it is usually obtained by conducting experiments and studying real speech. Finally, the rule labeled `<mycommand>` can be reused as a token in another rule definition.

While it would have been preferable for the ASM to implement a context-sensitive grammar, unique for each state of the system, this would have been resource-intensive and expensive in terms of switch delays.[13] Instead, a single global grammar was devised for use throughout the system. Figure 3.6 shows the ADA grammar definition implemented prior to user experiments. It instructs the system to accept long phrases, short commands, or dates and times. Long phrases, as defined in this grammar, may contain verbs, articles, pronouns, and other tokens. A short command is a one- or two-word utterance, intended to be used as a shortcut for advanced or hurried users. The format adopted for day-time schedule selection requires that both the weekday

[13]Trials using state-specific grammars showed a 5- to 10-second delay at each grammar change.

```
public <command> = ( [ /3/<longphrase> | /4/<shortphrase> | /2/<daytimephrase> ] );

<longphrase> = ( [<startphrase>] [<verb>] [<article>] <noun> [<endphrase>] );
<startphrase> = ( /4/I would like to | /4/I'd like to | /4/I want to | /1/may I | /2/can I | /2/can you );
<verb> = [please] ( /3/do | /4/see | /2/view | /3/look at | /4/make | /4/leave | /2.5/attach | /3/show me | /3/start |
                    /3/[re] record | /3/[re] play | /2/go | /4/choose | /4/confirm | /3/clear | /2/erase | /2/cancel );
<article> = ( /4/the | /4/a | /4/an | /1/Jeremy );
<noun> = ( /4/appointment | /1/meeting | /4/schedule | /4/message | /3/[available] documents | /4/written [ message | note ] |
           /4/video [ /5/message | /2/conference | /1/conferencing ] |
           /3/ ( /3/following | /3/next | /3/this | /1/preceding | /3/previous | /3/last )
           ( /3/week | /1/week's) [schedule] | /4/something else | /2/back | /2/home |
           /3/( /1/first | /1/second | /2/different | /2/another ) [ one | time ] | /2/note );
<endphrase> = ( /3/please | /2/now | /2/[ for | with ] Jeremy );

<shortphrase> = ( <yesno> | <writtenvideo> | <oneword> | <weekswitch> | <mainmenu> );
<yesno> = ( yes | no );
<writtenvideo> = ( written | video );
<oneword> = ( [re] record | [re] play | clear | erase | cancel | confirm );
<weekswitch> = ( /3/following | /3/next | /3/this | /1/preceding | /3/previous | /3/last ) [week];
<mainmenu> = ( schedule | message | documents );

<daytimephrase> = ( /2/<daytime>|/1/<timeday> );
<daytime> = [ /1/for | /2/on ] <day> [ at ] <time>;
<timeday> = [ at ] <time> [on] <day>;
<day> = ( /1/today | /1/tomorrow | /2/<weekday> );
<weekday> = ( /3/monday | /4/tuesday | /5/wednesday | /6/thursday | /7/friday );
<time> = [half past] (/4/<morningtime>|/6/<afternoontime>);
<morningtime> = ( eight | nine | ten | eleven ) [ oh clock | [thirty] a m ];
<afternoontime> = ( (twelve|noon) | one | two | three | four | five ) [ oh clock | [thirty] p m ];
```

Figure 3.6: ADA grammar definition in JSGF.

and time be spoken, in either order, within the same command.

The grammar in Figure 3.6 is provisional, though it is intended to evolve accordingly after studying users' speech patterns in experiments. In addition to weight adjustments, there will likely be rule changes along with the introduction of new words to accommodate a wider variety of vocabularies.

3.3.3 Sample Dialogue

In order to demonstrate the conversational abilities of the system, a sample interaction session is transcribed. The following dialogue describes a scenario in which a visitor walks in front of the ADA kiosk, instantly waking up the system from its idle state.

- ADA wakes up and greets the user.

ADA: *"Hmm."*

User: *"Can I look at the professor's schedule please?"*

- ADA shows the schedule for the current week.

ADA: *"Would you like to schedule an appointment?'*

User: *"Yes."*

- ADA asks the user to specify a time slot.

User: *"At three-thirty on Thursday."*

- ADA prompts the user to leave a written or video message.

User: *"I'd like to leave a video message."*

- User speaks into the camera and presses **Stop Recording** when finished.

User: *"Can you replay the message?"*

- ADA replays the recorded message for the user to review.

User: *"Confirm the appointment."*

- ADA confirms the appointment and returns to the home screen.

This transcript describes a dialogue between a visitor and the agent of the kiosk. The user in this scenario is issuing commands using different sentence structures. Initially, the visitor makes an interrogative request to look at the schedule. Later in the interaction, the user adopts a declarative tone when desiring to leave a video message. The final command to confirm the message is then spoken in the imperative. The grammar defined for the ASM implicitly supports all of these sentence constructs, accommodating a variety of users and moods.

3.4 PSI: Professor-Side Interface

Written entirely in PHP, the PSI is an application that serves as the back-end system to the ADA. Since the ADA machine is networked to a web server, the PSI can be accessed by entering its URL from any web browser. It communicates asynchronously with the ADA front-end through files, thus it functions independently regardless of whether the user-side is deployed.

3.4.1 Interface Overview

As shown in Figure 3.7, the PSI's web-based interface resembles the ADA front-end, borrowing its design themes and principles to ensure homogeneity. The avatar of the agent is present simply for ensuring aesthetic consistency between the systems.

The PSI evolved from a basic calendar system into a larger system with additional administrative functions and features. When viewing the calendar for the entire month or year, all the days containing events or appointments are highlighted.

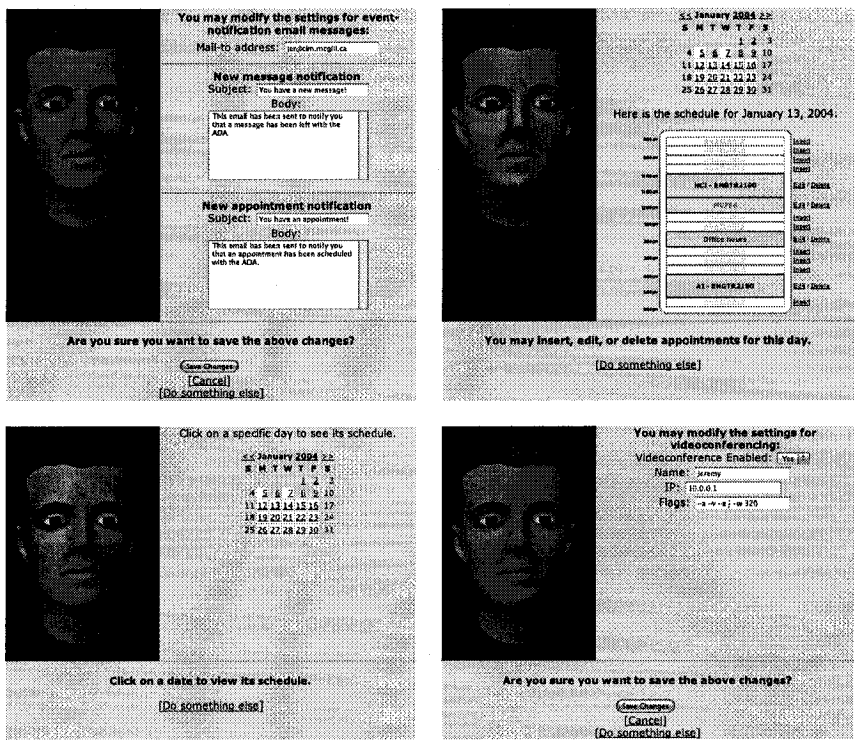

Figure 3.7: Screenshots of the PSI in different states.

Beyond the ability to insert and modify entries in the daily schedule, the system provides additional features that allow the professor to set the range of allowable appointment times and to manage recurring calendar events at daily and weekly intervals. Appointments set with the ADA kiosk are displayed in the PSI schedule, along with a hyperlink corresponding to the attached written note or video recording.

From the main page, the professor may view a listing of all messages left with the front-end and modify email notification settings. Additionally, parameters for the document browser and videoconferencing subsystems are editable through the PSI. The last link on the main page lets the professor access the kiosk activity log, providing a detailed chronological account of user actions performed with the front-end as well as all the spoken commands transcribed by the ASM.

3.4.2 Implementation Details

The rationale behind programming the PSI solely using PHP is based on the fact that server-side functionality is required to access and write to files on the network. Furthermore, avoiding client-side technologies such as Javascript is generally recommended to ensure maximal compatibility with all browsers and platforms. In terms of security, the system can be password-protected using basic .htaccess authentication.

The calendar code in the PSI is built from Cascade[14], and has been augmented with the ability to assign events within daily schedules and read data from files. The decision to store data in raw text files rather than in a database was to ensure that data can easily be shared with the front-end by adopting a common format and storing files in a shared path. Furthermore, since the files are saved in plain text format, they are manually readable without the PSI.

[14]http://www.cascade.org.uk/software/php/calendar/

Chapter 4

Empirical Study

"I have little respect for testing and evaluation in interface research. My argument, perhaps arrogant, is that if you have to test something carefully to see the difference it makes, then it is not making enough of a difference in the first place."

- NICHOLAS NEGROPONTE [18]

4.1 Experiment Design

Once the redesigned system was fully operational and deployed, we conducted a user study in order to observe and analyze interactions with the door attendant.[1] The objectives of the experiment were to observe users' tendencies and speech patterns, identify weaknesses in the system, determine the intuitiveness at different states in the interface, and to develop an understanding of which modalities are preferred by users.

The study compared the effects of different variables (modality, gender, native language, computer aptitude) on a variety of objective and subjective measures. Users interacted with the ADA and performed an assigned task given the following instructions:

[1]The required approval was obtained by the McGill University Research Ethics Board prior to recruiting subjects. The research compliance certificate is provided in Appendix C.

Condition	Modalities
Unimodal	Only speech (S)
	Only touch (T)
Multimodal	Both modalities (B)

Table 4.1: The three input modality combinations.

(a) *View the professor's schedule.*

(b) *Make an appointment for any available time slot.*

(c) *Use the stylus to write down your name on the notepad.*

(d) *Confirm the appointment when finished.*

The entire task, consisting of subtasks (a) through (d), was carried out unimodally and multimodally. The purpose of isolating the different input modalities for testing was to obtain qualitative and quantitative data indicating users' performance for each of the different trials. The experiment used a "within-subjects" design, meaning that each participant used the system in every modality combination (see Table 4.1). Thus, the assigned task was completed three times and the modality conditions were presented in counterbalanced order to reduce learning effects.

The order was randomized such that either both unimodal trials (S,T) were performed prior to the multimodal trial (B), or vice-versa. Thus, four possible modality orders were considered: (S→T→B), (T→S→B), (B→S→T), and (B→T→S). The rationale for considering these four orders (instead of all six) was to ensure that the multimodal case (B) was always either the first or last trial of the experiment. If the multimodal trial occurred first, then it was possible to gauge users' tendencies to adopt a modality without them having any prior experience using the system. Alternatively, if it occurred last, then users had acquired experience using the system with both modalities equally, and their tendencies were influenced accordingly.

4.2 Evaluation Procedure

Each participant was given a detailed consent form, describing their task in the experiment and the possible risks involved. After reading over and signing the form, they were handed a brief pre-experiment questionnaire (Appendix Figure D.1) to collect basic demographic information and to assess their computer experience. The instructions were repeated orally, and subjects were given an opportunity to ask any last-minute questions prior to starting.

Participants were randomly assigned one of the four modality orders in which to complete all three trials, and were requested to complete their assigned task in the shortest amount of time possible. In between each trial, subjects were allowed a thirty-second resting period before proceeding to the next trial. The setting of the experiment was, evidently, in the corridor outside the professor's office, and the written task instructions were posted on the wall next to the ADA kiosk for easy reference. For the entire duration of the experiment, the subjects were video-recorded unobtrusively in order to gather speech information and to observe their interactions with the touchscreen. In addition, all kiosk activities and events were recorded in the logfile.

Following the experiment, subjects were debriefed and given monetary compensation for their participation in the study. They were asked to sign the consent form once again, attesting that they were paid, and were reminded that their identities were to remain confidential. A post-experiment questionnaire (Appendix Figure D.2) was then handed to all participants in order to assess their opinions of the system, however approximately 15% of the tallied survey questions were either incomplete or unfilled.[2]

[2]Due to the non-compulsory nature of the survey, response data could have been skewed by those who chose to fill it out as opposed to those who didn't, therefore this was noted in its analysis.

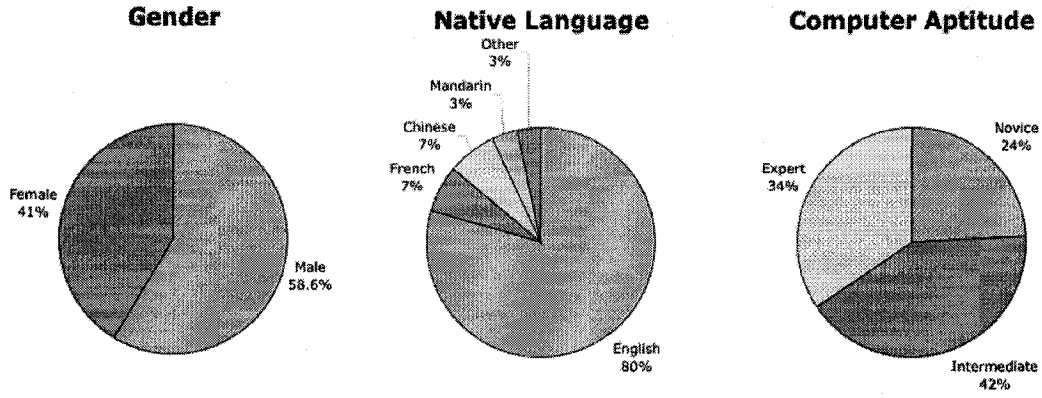

Figure 4.1: General statistics of the subjects.

4.3 Subject Demographics

Human subjects were recruited from the population of interest, consisting of McGill University students in their early to mid-twenties, resulting in a sample size of 29. As depicted in Figure 4.1, there were 12 female and 17 male participants, and English was the dominant native language. Although participants were not partitioned into groups, their level of computer proficiency was categorized according to the following taxonomy, as defined for our study:

- *Novice* - Rudimentary understanding of how to use a mouse/keyboard and a web browser.
- *Intermediate* - Average understanding of basic operating system features and how to use a word processor or office suite.
- *Expert* - Solid understanding of software and hardware and/or programming.

If subjects were unsure of which group they belonged to, then the experimenter recommended the appropriate category. While the expert subjects were almost exclusively recruited from within the Centre for Intelligent Machines (CIM) department, the remainder were selected from various faculties of the university.

4.4 Results and Analysis

The data analyzed in this section has been obtained from the post-experiment questionnaires, the generated ADA event log file, and the video-recorded interactions of subjects. The accurate transcription and annotation of video-recordings required careful observation of users' actions and utterances, as well as the synchronization of events with the ADA log file.

Statistical analysis techniques were employed in order to assess whether reported differences between conditions and between subjects could be wholly accounted for by error, or whether significant effects existed. Depending on the number of experimental conditions and the grouping of subjects, significance testing was used to obtain a p-value: the probability that the experimental result could have arisen randomly.

4.4.1 Task Completion

All of the participants successfully completed their assigned task three times (once for each of three modality restrictions: *speech-only*, *touch-only*, and *both modalities*). As described in Section 4.1, the task assigned to test subjects can be decomposed into four subtasks, whose execution times are defined below:

(a) *View the schedule* - The elapsed time between the commencement of the trial until the schedule option is successfully selected from the main menu.

(b) *Select a time slot* - The time required to pick one of the suggested appointment slots or manually select one from the timetable.

(c) *Leave a message* - The time taken to write a note with the stylus.[3]

(d) *Confirm the appointment* - The measured time from when the message has been completed until it is finally confirmed.

[3]The time required to leave a message was omitted from these results, as this depends on factors having no scientific relevance to this research (i.e. handwriting speed, name length, artistic effort). In practice, users are free to leave written notes or 30-second video messages.

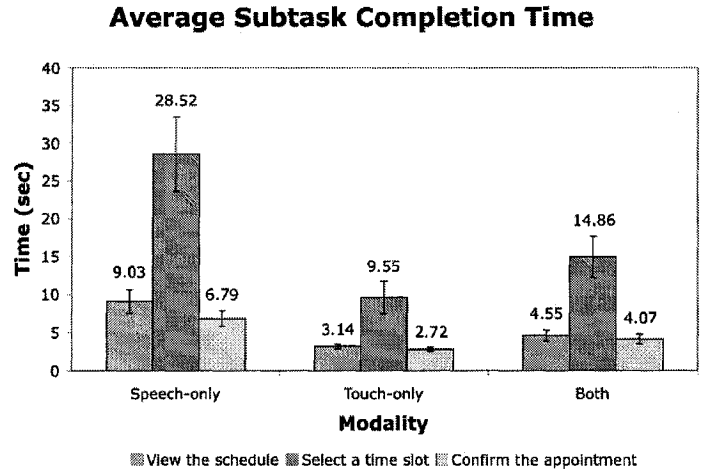

Figure 4.2: Average subtask completion time $[d]$ per modality $[i]$.

Subtask	χ_r^2	Friedman p	Page's L Trend p	ANOVA p
View the schedule	12.78	< 0.01	< 0.01	< 0.001
Select a time slot	20.76	< 0.001	< 0.001	< 0.01
Confirm the appointment	11.71	< 0.01	< 0.01	< 0.01

Table 4.2: Levels of significance for subtask completion time.

Figure 4.2[4] shows the average completion time for three subtasks in each modality condition. The subtasks are examined individually, and each is analyzed as a related design task for three conditions. The suggested statistical tests for such a scenario are the Friedman and Page's L trend tests (non-parametric), or the one-way related ANOVA (parametric) [8]. While the Friedman test is often used to determine overall

[4]The labels $[d][i]$ in the figure captions denote dependent and independent variables, respectively.

Subtask	Speech vs. Touch	Speech vs. Both	Touch vs. Both
View the schedule	< 0.01	< 0.01	Inconclusive
Select a time slot	< 0.01	< 0.05	Inconclusive
Confirm the appointment	< 0.01	< 0.05	Inconclusive

Table 4.3: Tukey HSD post-hoc levels of significance for subtask completion time.

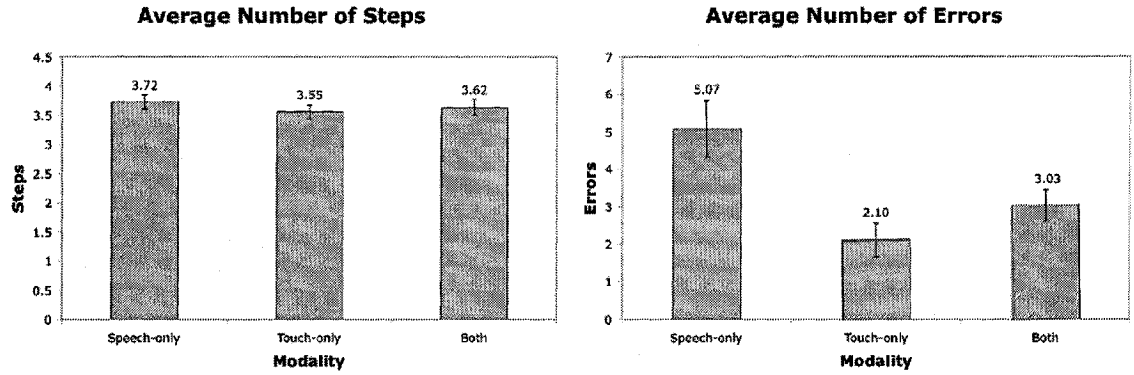

Figure 4.3: Average number of steps $[d]$ and errors $[d]$ measured per modality $[i]$.

Measure	χ_r^2	Friedman p	Page's L Trend p	ANOVA p
Number of steps	1.259	Inconclusive	< 0.50	< 0.47
Number of errors	10.40	< 0.01	< 0.05	< 0.01

Table 4.4: Levels of significance for number of steps and errors.

differences between conditions, Page's L trend test illustrates whether there is a trend in a particular order. The statistical analysis (see Table 4.2) shows that the differences between modalities are significant for subtask completion times, and that there is a significant trend between conditions, in order: touch-only, both, and speech-only (consistent across all three subtasks).

Appropriate post-hoc comparisons (Table 4.3) using Tukey's Honest Significant Difference (HSD) test shows that completing a task solely using speech is significantly slower than completing it multimodally or using only touch. There are many possible causes for such a substantial discrepancy between conditions, including the number of steps required and the propensity for errors for a given modality. The graphs in Figure 4.3 illustrate the average number of steps and errors measured for each modality trial.

Although it seems natural to think that successfully completing the task would entail performing a fixed number of steps, there is some variability in this figure due

to the fact that selecting a time slot can involve more than one step. A user may request that the ADA suggest some available times, or may simply pick a free slot in the schedule. Thus, the entire task can be performed in as little as three steps and up to as many as five steps (for either modality). As noted in Table 4.4, Friedman and Page L trend tests are inconclusive ($p > 0.05$) with respect to the number of steps, implying that the difference between modalities is insignificant. This is also apparent by observing the overlapping error bars in the first graph of Figure 4.3, indicating that the number of steps required to complete a task does not vary significantly between modalities. Furthermore, a one-way related ANOVA shows an insignificant difference between conditions, and demonstrates variance between subjects (F-ratio $= 2.86$).

Considering the nature of the modalities and their inherent propensity to errors, it is not surprising that speech input is susceptible to a greater number of errors than touch, as apparent in the second graph of Figure 4.3. The statistical analysis of the number of errors shows considerable differences ($p < 0.01$) between the three modality conditions. As in the case of subtask completion times, Page's L test indicates that there is an increasing trend with respect to the number of errors across the modalities, in order: touch-only, both, and speech-only. Thus, the similarity in trends between task completion times and number of errors can be partially due to the fact that users spend a lot of time recovering from errors.

4.4.2 Observed Errors

Errors, as defined in the context of this study, are incidents in which the immediate desired outcome is not realized. The granularity of an error pertains to individual events, such as speaking a command or pressing a button on the screen. In the case of touch input, a user might not properly position a finger within the bounds of the button (or release it in the same position). With speech input, the most common errors are due to incorrect command recognition. Although, for the most part with the ADA, the result of an erroneous input is a non-action, sometimes the result may

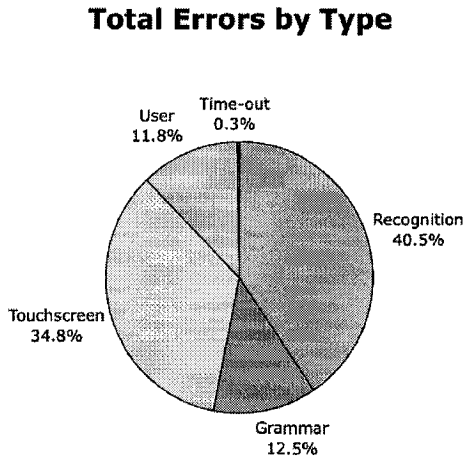

Figure 4.4: Breakdown of total measured errors.

be an undesired alternate action. Fortunately, the latter scenario seldom occurred during the course of experiments.

As depicted in Figure 4.4, errors are categorized as follows:

- *Recognition error* - The words uttered by the user are incorrectly recognized.
- *Grammar error* - The command is syntactically and logically valid (with respect to the immediate goal), yet not defined in the grammar.
- *Touchscreen error* - Touching the screen does not produce the desired outcome.[5]
- *User error* - The user has made the wrong selection or spoken the wrong command, inconsistent with the goal at hand.
- *Time-out error* - The user has exceeded the allotted time at a given state (usually one minute or longer) and the ADA has gone into "sleep" mode.

[5]When using the ADA, one will notice that the touchscreen is mounted in front of an LCD monitor, which results in a small gap between the two (approximately 2 inches). Despite the fact that the touchscreen is properly calibrated, the LCD screen and touchscreen positions do not necessarily correspond exactly due to a phenomenon known as parallax. Parallax is usually caused by the thickness of the touchscreen glass, but in the case of the ADA, it is amplified by the gap of air between the LCD screen and touchscreen.

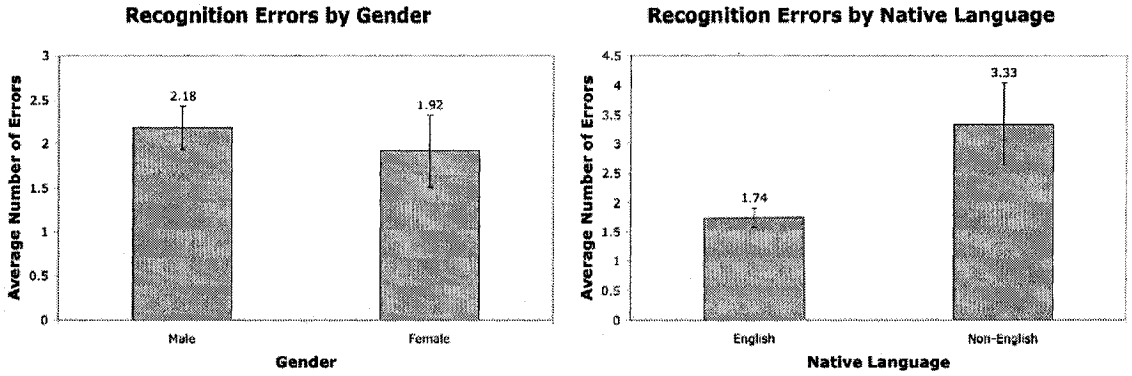

Figure 4.5: Recognition errors $[d]$ among genders $[i]$ and native languages $[i]$.

Measure	Mann-Whitney p	t test p
Recognition errors (Gender)	< 0.30	< 0.58
Recognition errors (Native language)	< 0.005	< 0.005

Table 4.5: Levels of significance for recognition errors.

As confirmed by the pie graph, it is expected that speech-related issues (recognition/grammar) account for the largest source of error. Naturally, recognition errors are unavoidable due to the fact that speech recognition engines are not perfect, and must deal with multiple accents and pronunciations. A variety of noise-canceling microphones exist, which can improve recognition accuracy and minimize the effect of background noise. Grammar errors, however, may be reduced by increasing the grammar's breadth to accommodate a larger dictionary of allowable words.

It is surprising to note that such a high number of touchscreen/parallax errors are measured (34.8%), though it is less of a problem for taller subjects due to the height at which the ADA is mounted on the wall. Also, since the button interface is based on "click" events, the input is only accepted once the user's finger is lifted from the screen. In addition to introducing a minor delay, computer novices who are unfamiliar with clicking a mouse may find this notion of raising their finger from the screen unintuitive. While it would be simple to replace "click" events with "mouse-down"

events, this could lead to undesired results caused by the parallax issues. Currently, users who miss their touch-target can rectify their action by guiding the mouse cursor above the desired point prior to releasing their finger. Ultimately, these issues may be avoided by replacing the current screen with an LCD panel with integrated direct-surface touch.

Returning to the issue of speech recognition errors, it is interesting to differentiate between different classes of speakers. Recognition performance between genders may be observed in Figure 4.5. The Mann-Whitney (non-parametric) and unrelated t tests (parametric) are employed to detect statistical significance between subject groups [8]. The overlapping standard error bars of the graph and the high p values in Table 4.5 indicate that the difference in recognition errors between genders is not meaningful. With regards to native language, however, the differences in recognition errors are statistically significant ($p < 0.005$), as expected. Measured recognition errors among non-native English-speaking users were (on average) over 90% higher than among anglophones.

4.4.3 Effects of Aptitude

The effect of subjects' computer proficiency on their overall performance is depicted in Figure 4.6. The results seem almost counterintuitive, but those self-proclaimed "expert" users are, on average, the worst performers in terms of task completion time and error rates. Those participants who described themselves as "novice" demonstrate the best results in almost every scenario. In order to verify the significance of the differences, Kruskal-Wallis (non-parametric) and one-way unrelated ANOVA tests (parametric) are used [8]. The p values in Table 4.6 demonstrate meaningful results in the speech-only condition, and post-hoc Tukey HSD tests are conclusive for the Novice vs. Expert gap throughout most of the conditions.

While it is difficult to speculate about why this is so, the results seem to imply

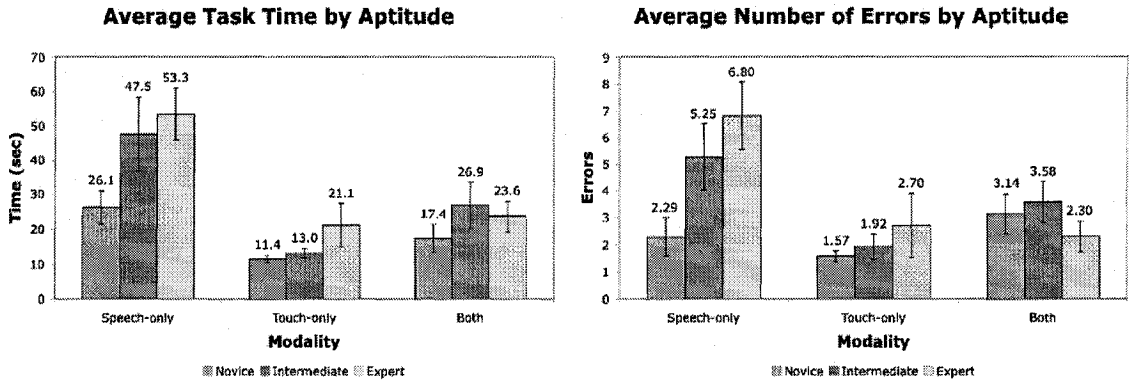

Figure 4.6: Average task time [d] and errors [d] among computer aptitude levels [i].

Measure	Kruskal-Wallis p	ANOVA p	Tukey HSD p
Time (Speech-only)	< 0.02	< 0.12	< 0.05 (Novice vs. Expert)
Time (Touch-only)	< 0.25	< 0.20	< 0.05 (Novice vs. Expert)
Time (Both)	Inconclusive	< 0.51	< 0.05 (Novice vs. Expert)
Errors (Speech-only)	< 0.05	< 0.05	< 0.05 (Novice vs. Expert)
Errors (Touch-only)	Inconclusive	< 0.61	Inconclusive
Errors (Both)	< 0.25	< 0.46	< 0.05 (Novice vs. Expert)

Table 4.6: Levels of significance for computer aptitude.

that the ADA's speech interface is significantly different from typical GUIs. The transfer of knowledge does not seem to apply to these expert users who are preconditioned to working with menus and cursors in order to complete a task. These results are encouraging because of the fact that computer novices are able to use the system without any preconceived notion of how the system works. Perhaps the conversational style of the ADA appeals more to beginners because advanced computer users are not conditioned to thinking about computers in this manner. While some may argue that these results signal a weakness in the system because its speech interface does not accommodate users of all levels equally, the results in Table 4.6 suggest that there is a "leveling of the playing field" throughout the touch-only and multimodal conditions.

Since the discrepancy is most apparent in the speech-only case, it could be suggested (from Figure 4.5) that perhaps many of the expert subjects were possibly non-English native speakers. However, this is not the case, since the number of non-English native subjects was proportional across all proficiency levels. Ultimately, these results demonstrate that, in the speech-only condition, performance varies inversely with computer aptitude. As for the touch-only and multimodal conditions, the figures are not statistically meaningful enough to conclude that there is a considerable difference in performance between subject groups.

4.4.4 Trial Sequence

Throughout this section, results have been analyzed according to modality as opposed to ordinality. When subjects were asked to perform their assigned task with various modality restrictions imposed, the order of these combinations was randomized in order to reduce learning effects. Thus, in Figure 4.7, the data is organized in terms of the sequentiality of the trials. As one should expect, the first trial yields the longest completion time, the highest number of steps, and the largest number of errors of all three trials. It would be logical to assume that the predictive capacity gained at each iteration would follow a progressive trend in performance across all three trials. While this is consistently true between the first two trials, the task completion time and error rates are slightly poorer in the third trial than in the second.

Statistical analysis of the means has been performed with Friedman and Page's L trend tests (non-parametric), as well as a one-way related ANOVA (parametric) [8], as shown in Table 4.7. The p values yielded by the tests are inconclusive ($p > 0.05$) with respect to any significant transfer of knowledge or learning rate between trials.

As represented in Table 4.1, the four counterbalanced condition orders are divided into two groups:

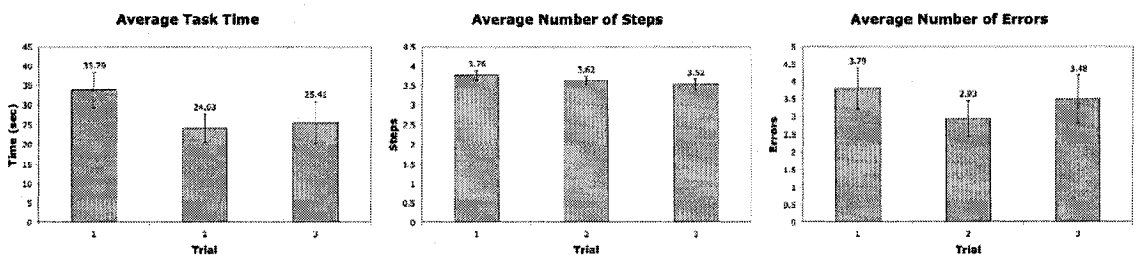

Figure 4.7: Average task time $[d]$, steps $[d]$, and errors $[d]$ per iteration $[i]$.

Measure	χ_r^2	Friedman p	Page's L Trend p	ANOVA p
Task time	4.207	< 0.20	< 0.20	< 0.28
Number of steps	2.138	< 0.50	< 0.30	< 0.22
Number of errors	0.431	Inconclusive	< 0.50	< 0.63

Table 4.7: Levels of significance for trials.

(1) unimodal→multimodal (e.g. S→T→B and T→S→B)

(2) multimodal→unimodal (e.g. B→S→T and B→T→S)

On average, subjects performed 73% of steps using the touchscreen when interacting multimodally. Depending on which trial sequence was assigned, this figure may have varied if the multimodal trial was performed before or after the unimodal trials. The average percentages of steps performed with speech during the multimodal trial are 29% and 22%, for cases (1) and (2) respectively. Although there is little discrepancy between the two figures (Mann-Whitney and t tests yield $p > 0.10$), the propensity to use speech in the latter case is slightly lower. From these figures, it can be speculated that participants are initially shy or skeptical about speaking to the ADA, but are more willing to once they have been exposed to communicating to it with speech.

State	Most Common Responses	Freq.
0	"[may i|can i|could i|i want to|i'd like to] (see|view) [jeremy's|the|the professor's] schedule [please|now]?"	60%
	"[jeremy's] schedule"	23%
2	"[yes] [can i|could i|i'd like to|i want to|i would like to] (make|schedule) an appointment [now|please]"	38%
	"yes|no"	35%
	"[no] [i'd like to] choose another time"	13%
	"[i'd like to see|see] the following week"	5%
2-5	"[for] <day> [at] <time>"	49%
	"[for|on] <day>"	9%
	"<time>"	6%
	"<time> [for] <day>"	6%
6	"confirm appointment"	44%
	"[i'd like to] confirm [now|please]"	40%
	"(i'll | i would like to | i'd like to | i want to) confirm the appointment"	10%

Table 4.8: Summary of spoken commands issued to the ADA during experiments.

4.4.5 Speech Input Summary

All verbal commands issued to the Automated Door Attendant have been grouped according to the grammatical structure of the phrases spoken (refer to Table 4.8[6]). The objective is to gain an understanding of how users adapt their language when interacting with the system. It is necessary to examine the propensities of users to speak in short phrases or full sentences, and whether their responses are influenced by the wording of system prompts.

Observing the transcripts, it is noticed that commands are issued in short phrases and full sentences in roughly equal ratios. Additionally, 32% of spoken commands are exact literal readings of menu options from the screen, such as *"See Jeremy's schedule"* or *"Make an appointment"*. For time-slot selection, nearly 45% of responses are

[6]For descriptions of states, refer to Table B.1.

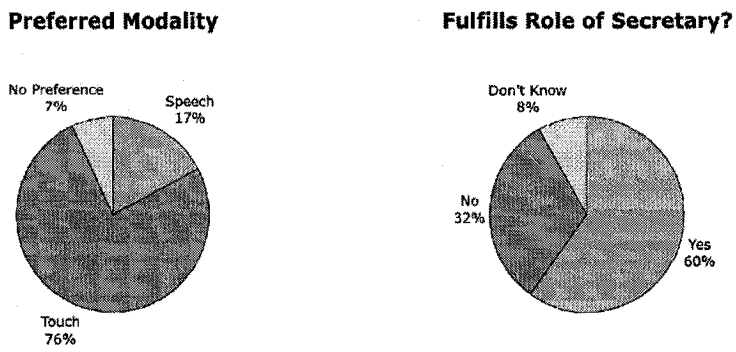

Figure 4.8: Results of the post-experiment questionnaire.

found to be incomplete or incorrectly phrased. The system requires that the day of the week and the time be uttered in any order within the same phrase, however, it is often the case that users fail to provide the required details. This implies the need for the system to present the user with clearer instructions when choosing an appointment slot from the schedule.

The prosody and speed of users' speech is often altered when speaking to the ADA. Unfortunately, there were no available metrics or comparisons upon which to base this claim, therefore it is difficult at this point to ascertain this incidence objectively.

4.4.6 Survey Data

Following the experiment, a survey was issued to all subjects in order to obtain additional qualitative feedback. Users were instructed to answer a brief questionnaire assessing their experience with the different aspects of the system, such as the interface and the perceived accuracy of the speech recognition. On average, participants responded to 85% of the questions in the survey, therefore the response data is of limited accuracy.

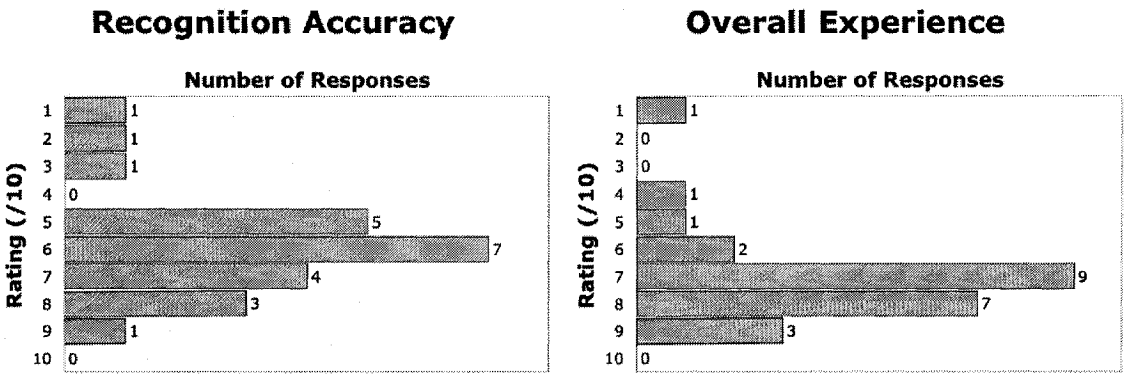

Figure 4.9: Results of the post-experiment questionnaire (continued).

Initially, users were asked which modality they preferred using and also whether they think that the ADA adequately fulfills the role of a secretary (Fig. 4.8). The majority of responses indicate that touch is the preferred modality (76%), whereas speech is only preferred by 17% of test subjects. These percentages are consistent with the tendencies observed earlier, implying that users will employ their preferred modality when given the choice.

The questionnaire also asked users to rate their perceived accuracy of the speech recognition engine and to give a general rating of their overall experience using the system, both on an integer scale of 1-10. The respective mean scores obtained are 5.8 for recognition accuracy and 7.0 for overall experience (see Figure 4.9). These standalone subjective measures would have been more valuable if participants had been asked to compare the ADA to another system, thus they only provide a rough idea of users' opinions.

Participants were also asked to note some of the characteristics of the ADA that they found the most and the least appealing. The gathered responses are compiled visually in Figure 4.10. The positive responses do not yield surprising results, however, the negative replies indicate a strong dissatisfaction with the quality of the speech recognition engine. While some of the issues may be addressed (*Limited grammar,*

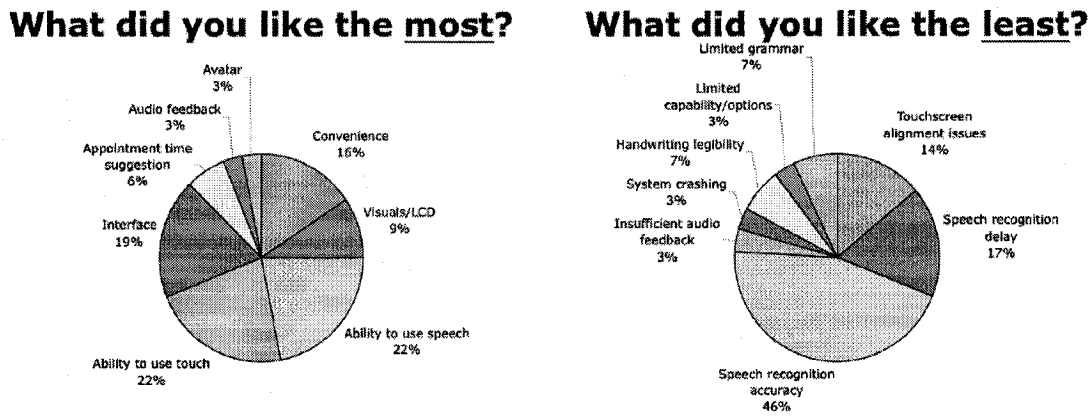

Figure 4.10: Results of the post-experiment questionnaire (continued).

System crashing, Insufficient audio feedback, Stylus handwriting legibility), others will be more difficult to resolve (*Touchscreen alignment, Speech recognition delay*).

4.5 Discussion

Comparing the results of the ADA empirical study to those of similar systems makes it possible to interpret the data further. In Office Monitor [31], Yankelovich had noted that interaction times with a human secretary were generally limited to 10-20 seconds. With the Automated Door Attendant, the average task completion time is 30-60 seconds[7], which is comparatively shorter than the times measured with the Office Monitor prototype (20-120 seconds).

Some surprising results were observed when users interacted multimodally. On average, task completion times and number of errors were higher during multimodal interaction than in unimodal touch-only interaction. In most other studies, such as August [10] and MASK [13], subjects generally performed better when they were allowed to mix modalities freely. However, it is generally accepted that speech is slower

[7]This is assuming that it takes 15 seconds to leave a written or video message.

and prone to more errors than touch [30] and, as Oviatt [23] states, a multimodal system is not necessarily more efficient than a unimodal one. Nevertheless, multimodality does offer the potential for greater flexibility and ease of use.

Another unexpected trend was noticed when results were sorted according to trial order. While not statistically significant, the data showed that users' performance improved in the second trial, but then worsened in the third trial. In the MASK [13] study, for example, task times decreased uniformly across trials.

The most significant source of errors experienced by users were related to speech recognition quality. Consequently, this impacted task completion times since subjects would have to repeat their commands, often more than once. Thus, it is possible to see the potential benefit of resorting to a Wizard-of-Oz approach for future user experiments. Perhaps it could have been interesting to combine both testing methodologies by splitting subjects into two groups (Wizard-of-Oz and automated speech recognition) and then observing the resulting effects.

Further data can be collected by conducting an unsupervised study of unsuspecting visitors, as described in the Office Monitor [31] and August [9] experiments. Although ethical considerations may need to be determined prior to undertaking such a project, it would surely provide additional valuable data.

Chapter 5

Future Directions and Conclusions

"That people interact differently with each other than they do with computers is clear. Many believe that by changing how we interact with machines to resemble more how we interact with people, we will result in some kind of HCI panacea."

- Bill Buxton [2]

5.1 Future Directions

Work is currently underway to augment the ECA with animated movements and non-verbal expressions. By introducing computer vision techniques, the 3D head could orient itself with respect to the user's position and provide gaze awareness with eye tracking. This can be taken even further by implementing a text-to-speech subsystem and synchronizing the output with the agent's lips.

Possible additions to the system could include Bluetooth or infrared for syncing appointments with PDAs. The ability to recognize repeat visitors may be realized using voice or face recognition software, or alternatively by implementing a magnetic card reader for reading student ID cards.

Future proposed improvements should be in consequence to the empirical study results and the feedback received from test subjects. Some of these changes, proposed

in the previous chapter, are related to the reliability of the recognition engine and the comprehensiveness of the defined grammar. Other changes may require more dramatic measures to implement them. Ultimately, many of the study's findings deserve further investigation in future undertakings of this project.

5.2 Conclusions

The work outlined in this thesis represents a public kiosk system providing an affordable alternative to a secretary. Implemented as a multimodal system, it provides users with a simulated conversational interface to perform tasks. While its current application is limited to a door attendant, the fundamental model of the ADA can serve as a basis for many applications that require a natural-language interface to make complex systems more usable.

The main contributions include the analysis of existing systems and literature to explore multimodal dialogues in several different contexts, as well as the detailed design and usability evaluation of the ADA system. The results of the empirical study serve to illustrate the tendencies of users and to expose the strengths and weaknesses of the system.

One of the issues revealed in the discussion of experiment results is the surprising performance of novice users with respect to advanced computer users. The data shows that beginners were able to complete the assigned task using speech in less time while encountering fewer errors than other subjects. This raises many questions, particularly with respect to the control dynamics of conversational interfaces contrasted to interfaces used in the computer applications that advanced subjects are accustomed to using. When interacting with an ECA, particularly with speech, the user is delegating tasks to an intermediary and making requests. Whether speaking in an interrogative, declarative or imperative tense, the scenario is the same: the agent

is being asked to do something on the user's behalf. This may seem counterintuitive to someone experienced with modern operating systems, because it is generally the user who possesses the control.

Human-human interaction employs various means of communication beyond spoken language, such as gestures, and facial expressions. These aspects of communication provide forms of entrainment that capture the attention of participants in a conversation. Implementing embodied conversational agents in interfaces should take advantage of these nonverbal cues in order to provide a richer dialogue between humans and computers.

In the context of multimodal HCI, there is still much research required to define guidelines for determining which modalities are best suited for various purposes. This will require plenty of inter-disciplinary research between experts in psychology, anthropology, and human-computer interaction to resolve all of the unanswered questions.

REFERENCES

[1] D. Bancroft and M. Wyse, *The Automated Door Attendant*, ECSE-494 Project Lab Report, Montreal, Quebec: 2001.

[2] W. Buxton, *Speech, Language & Audition*, Readings in Human Computer Interaction, Morgan Kaufmann Publishers, San Francisco, California: 1995.

[3] J. Cassell and T. Stocky, *MACK: Media lab Autonomous Conversational Kiosk*, Proceedings of Imagina'02 (Monte Carlo, 2002), Cambridge, Massachusetts: 2002.

[4] J. Cassell, *More Than Just Another Pretty Face: Embodied Conversational Interface Agents*, Communications of the ACM 43(4): 70-78, Cambridge, Massachusetts: 2000.

[5] V.H. Denenberg, *Statistics and Experimental Design for Behavioral and Biological Researchers*, John Wiley & Sons, New York, NY: 1976.

[6] S. DiPaola, *Facade: Stanford Facial Animation System*, `http://www.dipaola.org/stanford/facade/`, Stanford, California: 2001.

[7] J.L. Gauvain and J.J. Gangolf, *Speech Recognition for an Information Kiosk*, Proceedings of International Conference on Spoken Language Processing '96 (Philadelphia, 1996), Spoken Languages Processing Group, Orsay, France: 1996.

[8] J. Greene and M. d'Oliveira, *Learning to Use Statistical Tests in Psychology*, Open University Press, Buckingham, UK: 1999.

[9] J. Gustafson, N. Lindberg and M. Lundeberg, *The August Spoken Dialogue System*, Proceedings of Eurospeech'99, Budapest, Hungary: 1999.

[10] J. Gustafson, L. Bell, *Interaction with an Animated Agent in a Spoken Dialogue System*, Proceedings of Eurospeech'99, Budapest, Hungary: 1999.

[11] M. Johnston and S. Bangalore, *MATCHkiosk: A Multimodal Interactive City Guide*, Association of Computational Linguistics (ACL2004), Barcelona, Spain: 2004.

[12] T. Koda, and P. Maes, *Agents with Faces: The Effects of Personification of Agents*, Fifth IEEE International Workshop on Robot and Human Communication (RO-MAN'96), Tsukuba, Japan: 1996.

[13] L. Lamel and S. Bennacef, *User Evaluation of the MASK Kiosk*, Speech Communication (Sep 2002), Paris, France: 2003.

[14] J.T. Luftig and V.S. Jordan, *Design of Experiments in Quality Engineering*, McGraw Hill, New York, NY: 1998.

[15] M. C. Maguire, *A Review of User-Interface Design Guidelines for Public Information Kiosk Systems*, International Journal of Human-Computer Studies (v.50 n.3), Loughborough, Leics: 1999.

[16] E. Makinen and S. Patomaki, *Experiences on a Multimodal Information Kiosk with an Interactive Agent*, Nordic Forum for Computer-Human Interaction (Arhus, Denmark, 2002), Tampere, Finland: 2002.

[17] C. Min, *Multimodal User Interface Research (MUIR) Model, Specification and Design*, Peking University, Beijing, China: 1997.

[18] N. Negroponte, *Being Digital*, Random House Inc., New York, New York: 1995.

[19] J. Nielsen, *Usability Engineering*, AP Professional Press, Boston, Massachusetts: 1994.

[20] D. Norman, *The Psychology of Everyday Things*, Perseus Publishing, Boston, Massachusetts: 1988.

[21] D. Norman, *Why Interfaces Don't Work: The Art of Human-Computer Interface Design*, Addison-Wesley, Reading, Massachusetts: 1990.

[22] S. Oviatt, A. DeAngeli and K. Kuhn, *Integration and Synchronization of Input Modes during Multimodal Human-Computer Interaction*, Proceedings of Conference on Human Factors in Computing Systems (CHI'97), 1997.

[23] S. Oviatt, *Ten Myths of Multimodal Interaction*, Communications of the ACM (Vol. 42, No. 11), 1999.

[24] R. Raisamo, *Evaluating Different Touch-Based Interaction Techniques in a Public Information Kiosk*, Conference of the Computer Human Interaction Special Interest Group of the Ergonomics Society of Australia (1999), Tampere, Finland: 1999.

[25] R. Raisamo, *Multimodal Human-Computer Interaction: A Constructive and Empirical Study*, University of Tampere, Tampere, Finland: 1999.

[26] J. Raskin, *The Humane Interface: New Directions for Designing Interactive Systems*, Addison-Wesley Professional, Reading, Massachusetts: 2000.

[27] A. Saifee, *Automated Door Attendant: Speech Integration*, ECSE-494 Project Lab Report, Montreal, Quebec: 2004.

[28] S. Steininger, *Human-Computer Communication: Wizard Of Oz-Experiments In SmartKom*, Conference of the Society for Cognitive Science, Leipzig, Germany: 2001.

[29] W. Wahlster, N. Reithinger and A. Blocher, *SmartKom: Towards Multimodal Dialogues with Anthropomorphic Interface Agents*, Proceedings of International Status Conference "Human Computer Interaction", Berlin, Germany: 2001.

[30] N. Yankelovich and G.A. Levow, *Designing SpeechActs: Issues in Speech User Interfaces*, Proceedings of Computer-Human Interaction (1995), Chelmsford, Massachusetts: 1995.

[31] N. Yankelovich and C.D. McLain, *Office Monitor*, Proceedings of Computer-Human Interaction (April 1996), Conference on Human Factors in Computing Systems, Vancouver, British Columbia, Canada: 1996.

Appendix A

Comparison with Previous System

Although our design was partly influenced by previous implementations [1] of the system in terms of functional requirements, the new interface is significantly different. Many of the lessons learned from evaluating the previous system led to design improvements for the current version. The screenshots in Figure A.1 compare the main menu, schedule, and video-recording states between the previous and current systems.

Observing the interface of the previous system, it is immediately noticeable that this is a Windows application due to the visible title bar at the top and the recognizable GUI widgets found on the screen. The menu selections are inside of a list box, an inappropriate choice for a touchscreen. The box in the bottom left corner displaying the command heard by the system is a text entry box, another confusing choice of controls. Lastly, the avatar at the top left of the screen is not very engaging to the user, despite its animated expressions.

Our design introduces many modifications derived from criticisms of the old system. Among these changes is the full-screen mode of the interface and the reticence to use generic GUI controls that would hint to the visitor that they are using a computer program. In addition to eliminating the look and feel of an application, the maximization of screen real estate allocation helps ensure that the size of clickable elements is equal to that of a typical finger on the touch-screen.

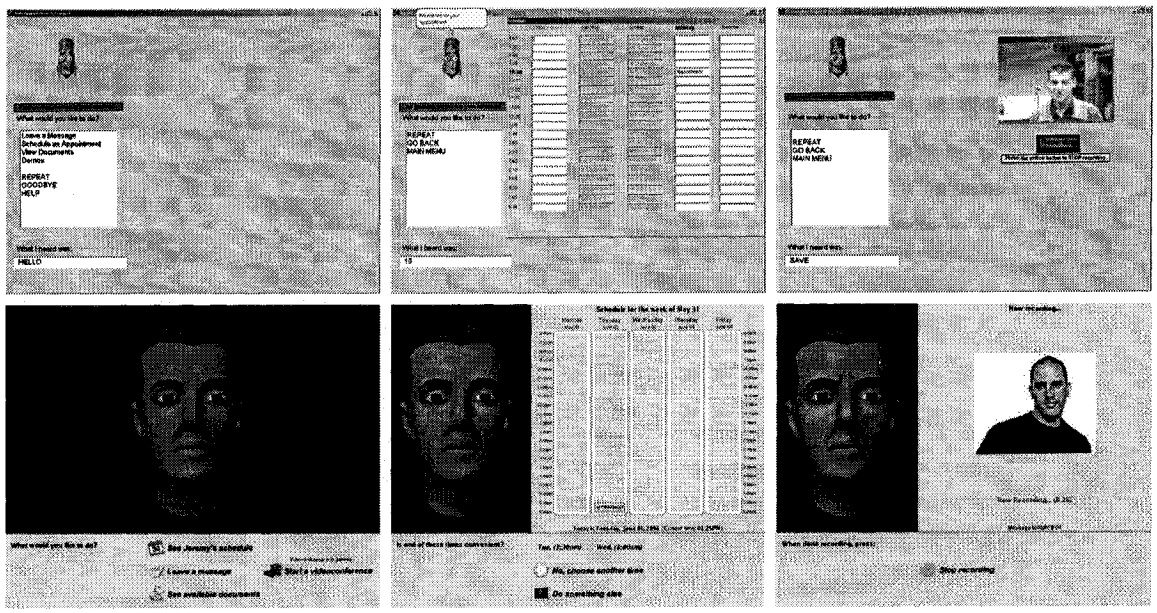

Figure A.1: Screenshots of the previous (above) and current (below) systems.

The decision to use an anthropomorphic agent, occupying over 25% of the screen's area, is an extension of the virtual secretary metaphor, entraining the user to shift their focus away from the "interface" and towards the task.

Lastly, the underlying hardware components have been upgraded to meet the heightened requirements of the redesigned ADA. The older Windows PC is superseded by a more powerful machine running Linux, and the CRT monitor is replaced with an LCD screen.

Appendix B

State Transition Table

The following table describes the internal state representation of the ADA front-end.

State	Internal State Name	Next State(s)
0	MAINMENU	1,2,12,19,20
1	SCREENSAVER	0
2	SCHEDULE_THISWEEK	0,3,4,6
3	SCHEDULE_NEXTWEEK	0,2,4,6
4	SCHEDULE_SUGGEST	0,5,6
5	SCHEDULE_MANUALSELECT	0,6
6	SCHEDULE_APPOINTMENT_NOTEPAD	0,5,7,11
7	SCHEDULE_APPOINTMENT_RECORD	0,8
8	SCHEDULE_APPOINTMENT_RECORDING	9
9	SCHEDULE_APPOINTMENT_STOPPED	0,8,10,11
10	SCHEDULE_APPOINTMENT_REPLAYING	0,8,11
11	SCHEDULE_APPOINTMENT_CONFIRM	0
12	MESSAGE_SELECT	0,13,14
13	MESSAGE_NOTEPAD	0,18
14	MESSAGE_RECORD	0,15
15	MESSAGE_RECORDING	16
16	MESSAGE_STOPPED	0,15,17,18
17	MESSAGE_REPLAYING	0,15,18
18	MESSAGE_CONFIRM	0
19	DOCUMENTS_VIEW	0
20	VIDEOCONF_START	0

Table B.1: State transition table.

Appendix C

Research Compliance Certificate

The following certificate was provided by Lynda McNeil, Research Ethics Officer.

Appendix D

Experiment Questionnaires

The following documents are excerpts from the forms provided to experiment subjects.

Gender: [ ] Male [ ] Female

Native language: [ ] English [ ] French [ ] Other: _____

Level of computer experience: [ ] Beginner [ ] Intermediate [ ] Advanced

Figure D.1: Pre-experiment questionnaire.

Which input modality did you prefer using?

[ ] Speech [ ] Touch-screen [ ] No preference

In your opinion, do you feel that the Automated Door Attendant adequately fulfills its role as a "virtual secretary"?

[ ] Yes [ ] No

On a scale of 1-10, how would you rate the quality of the speech **recognition**? ____/10 (where 1 is the lowest score, and 10 is the highest)

On a scale of 1-10, how pleasant was your experience using the ADA? ____/10

Which aspects of the ADA did you like the **most**?

Which aspects of the ADA did you like the **least**?

Figure D.2: Post-experiment questionnaire.

Appendix E

Statistical Tables

The following tables are provided to complement the graphs in Chapter 4. Please note the following acronyms used throughout this appendix:

S/T/B: Speech-only / Touch-only / Both modalities

M/F: Male / Female

E/N-E: English / Non-English

N/I/E: Novice / Intermediate / Expert

SS: Sums of squares

df: Degrees of freedom

MS: Mean squares

F: F-ratios

Figure	Description	N	Mean	Std.Dev.	Std.Error
Fig. 4.2	View the schedule (S)	29	9.03	8.38	1.56
	View the schedule (T)	29	3.14	1.64	0.30
	View the schedule (B)	29	4.55	3.73	0.69
Fig. 4.2	Select a time slot (S)	29	28.52	26.77	4.97
	Select a time slot (T)	29	9.55	11.46	2.13
	Select a time slot (B)	29	14.86	14.54	2.70
Fig. 4.2	Confirm appointment (S)	29	6.79	5.71	1.06
	Confirm appointment (T)	29	2.72	1.53	0.28
	Confirm appointment (B)	29	4.07	3.59	0.67
Fig. 4.3	Number of steps (S)	29	3.72	0.65	0.12
	Number of steps (T)	29	3.55	0.63	0.12
	Number of steps (B)	29	3.62	0.73	0.14
Fig. 4.3	Number of errors (S)	29	5.07	4.04	0.75
	Number of errors (T)	29	2.10	2.43	0.45
	Number of errors (B)	29	3.03	2.21	0.41
Fig. 4.5	Number of errors (M)	17	2.18	1.04	0.25
	Number of errors (F)	12	1.92	1.43	0.41
Fig. 4.5	Number of errors (E)	23	1.74	0.78	0.16
	Number of errors (N-E)	6	3.33	1.72	0.70
Fig. 4.6	Task time (N) (S)	7	26.14	12.69	4.80
	Task time (N) (T)	7	11.43	2.70	1.02
	Task time (N) (B)	7	17.43	10.50	3.97
Fig. 4.6	Task time (I) (S)	12	47.50	37.13	10.72
	Task time (I) (T)	12	13.00	4.59	1.33
	Task time (I) (B)	12	26.92	23.11	6.67
Fig. 4.6	Task time (E) (S)	10	53.30	24.04	7.60
	Task time (E) (T)	10	21.10	19.68	6.22
	Task time (E) (B)	10	23.60	13.99	4.42

Table E.1: Table of means and standard deviations.

Figure	Description	N	Mean	Std.Dev.	Std.Error
Fig. 4.6	Number of errors (N) (S)	7	2.29	1.89	0.71
	Number of errors (N) (T)	7	1.57	0.53	0.20
	Number of errors (N) (B)	7	3.14	1.95	0.74
Fig. 4.6	Number of errors (I) (S)	12	5.25	4.33	1.25
	Number of errors (I) (T)	12	1.92	1.62	0.47
	Number of errors (I) (B)	12	3.58	2.64	0.76
Fig. 4.6	Number of errors (E) (S)	10	6.80	4.02	1.27
	Number of errors (E) (T)	10	2.70	3.77	1.19
	Number of errors (E) (B)	10	2.30	1.77	0.56
Fig.4.7	Task time (1)	29	33.79	24.45	4.54
	Task time (2)	29	24.03	18.79	3.49
	Task time (3)	29	25.41	28.17	5.23
Fig.4.7	Number of steps (1)	29	3.76	0.636	0.118
	Number of steps (2)	29	3.62	0.622	0.115
	Number of steps (3)	29	3.52	0.738	0.137
Fig.4.7	Number of errors (1)	29	3.79	3.19	0.592
	Number of errors (2)	29	2.93	2.72	0.506
	Number of errors (3)	29	3.48	3.73	0.692

Table E.2: Table of means and standard deviations (continued).

Source of Variance	*SS*	*df*	*MS*	*F*
Modality	549.68	2	274.84	9.73
Subjects	790.60	28	28.24	0.96
Error	1638.99	56	29.27	
Total	2979.26	86		

Source of Variance	*SS*	*df*	*MS*	*F*
Modality	5552.09	2	2776.05	7.03
Subjects	11064.62	28	395.17	1.19
Error	18595.24	56	332.06	
Total	35211.95	86		

Source of Variance	*SS*	*df*	*MS*	*F*
Modality	249.26	2	124.63	8.11
Subjects	430.34	28	15.37	0.95
Error	910.07	56	16.25	
Total	1589.68	86		

Table E.3: One-way ANOVA (related) tables for Table 4.2.

Source of Variance	*SS*	*df*	*MS*	*F*
Modality	0.44	2	0.22	0.28
Subjects	22.23	28	0.79	2.86
Error	15.56	56	0.28	
Total	38.23	86		

Source of Variance	*SS*	*df*	*MS*	*F*
Modality	133.40	2	66.70	7.93
Subjects	235.59	28	8.41	0.90
Error	523.93	56	9.36	
Total	892.92	86		

Table E.4: One-way ANOVA (related) tables for Table 4.4.

Source of Variance	*SS*	*df*	*MS*	*F*
Computer aptitude	3762.46	2	1881.22	2.29
Error	21337.96	26	820.69	
Total	25100.42	28		

Source of Variance	*SS*	*df*	*MS*	*F*
Computer aptitude	495.46	2	247.73	1.71
Error	3760.61	26	144.64	
Total	4256.08	28		

Source of Variance	*SS*	*df*	*MS*	*F*
Computer aptitude	426.47	2	213.24	0.668
Error	8299.03	26	319.19	
Total	8725.50	28		

Source of Variance	*SS*	*df*	*MS*	*F*
Computer aptitude	96.78	2	48.39	3.37
Error	373.28	26	14.36	
Total	470.06	28		

Source of Variance	*SS*	*df*	*MS*	*F*
Computer aptitude	6.15	2	3.076	0.50
Error	158.73	26	6.11	
Total	164.88	28		

Source of Variance	*SS*	*df*	*MS*	*F*
Computer aptitude	7.82	2	3.91	0.80
Error	127.87	26	4.92	
Total	135.70	28		

Table E.5: One-way ANOVA (unrelated) tables for Table 4.6.

Source of Variance	SS	df	MS	F
Trial order	1617.68	2	808.84	1.52
Subjects	14891.10	28	531.83	0.88
Error	33959.66	56	606.42	
Total	50468.44	86		

Source of Variance	SS	df	MS	F
Trial order	0.85	2	0.43	0.54
Subjects	22.23	28	0.79	2.93
Error	15.15	56	0.27	
Total	38.23	86		

Source of Variance	SS	df	MS	F
Trial order	11.06	2	5.53	0.66
Subjects	235.59	28	8.41	0.73
Error	646.28	56	11.54	
Total	892.92	86		

Table E.6: One-way ANOVA (related) tables for Table 4.7.