CHAPTER 1 Introduction

Motivation for This Research -A Need for Agent-based Interfaces

Software agents

The Internet provides us with enormous amounts of information, and it is involving more people in the world of bits. Computers are becoming essential tools for entertainment, socializing, as well as getting information and news. Improvements in computer technology allow us to have access to information from anywhere at anytime. We need help in dealing with this information overload to separate information from noise. Software agents are computer programs that provide assistance to a user dealing with computer based applications, such as information filtering, meeting scheduling, selecting music, etc. Software agents are different from current programs in three ways: 1) they are **proactive**--they initiate actions, 2) they are **adaptive**--they learn from the user's preference and habits, 3) they are **personalized**--they change their way of helping the user according to what they have learned about

the user [Maes 95a]. Software agents work autonomously without waiting for a user's command. The user and the agent both communicate to each other to complete a task. As we depend more on computers to complete our daily tasks, we will have to rely for some part of our productivity on software agents, or "digital assistants." As Maes [Maes 94] says, we evaluate such digital assistants using two criteria: their ability to perform a task on behalf of the user, and their trustworthiness, or the assurance that allows the user to delegate tasks to an agent without worrying. We will only adopt agents if we can trust them and control their actions.

Agents assist users in various ways and domains. The tasks suited for computer agents are the same as those to which we employ human agents: tasks that require expertise, skill, resources, or labor to accomplish some goal, and tasks that we don't want to do or cannot do ourselves. Laurel and Maes categorize those applications as follows [Laurel 90, Maes 94]:

- **information**: navigation and browsing, information retrieval, sorting and organizing, filtering
- **work**: reminding, programming, scheduling, advising, managing mail
- **learning**: coaching, tutoring, providing help
- **entertainment**: playing against, playing with, performing.

Agent-based Interface

Despite the increasing interest in software agents, very little research on the representation of agents has been done. The dominant user interface is still GUI with direct manipulation, which was introduced in the '70s [Goldberg 84]. GUIs have been refined over the past 20

years to make them more usable [Norman, D. 88], but the original idea of using a direct manipulation metaphor has not been changed. As Lieberman says [Lieberman 96b], a direct manipulation interface provides a user high degree of control and the user can initiate tasks when the interface is simple enough to understand intuitively. As software becomes more powerful, the supporting interface becomes more complex. Consequently, the direct manipulation interface no longer provides high degree of control.

Since software agents change the way we interact with computers, there is a need for building a new interface to support software agent applications. Negroponte suggests a new paradigm of the “social” interface. He writes, “What we today call ‘agent-based interface’ will emerge as the dominant means by which computers and people talk with one another.... the interface will need size, shape, color, tone of voice, and all the other sensory paraphernalia” [Negroponte 95]. This “agent-based” interface should support completing cooperative tasks based on the user’s delegation of tasks to agents. Thus, the representation of software agents will be an important issue for agents’ implementation. Maes addresses the important issues that need to be resolved for building a successful software agent [Maes 94]:

- How the agent “does the right thing” to assist users with their tasks.
- How we guarantee that users feel comfortable delegating tasks to the agent.

Most previous research in agents has focused on the former. The latter issue has not been studied much in agent research. Maes [Maes 94] lists several remaining issues in software agents research among which are:

- What is the best metaphor for software agents?

- What characteristics make an agent trustworthy and intelligent?
- Should agents use facial expressions and other means of personification?

With few exceptions (see Chapter 2 for discussion), the bulk of HCI research has not been on personified interfaces, but on direct manipulation interfaces. The most commonly used interface for personification is a human face. However, it is still an open question whether software agents should be anthropomorphized in the interface. Several arguments against the use of agents have been offered. For example, Lanier argues that agents make users lazy and narrow-minded [Lanier 95]. Shneiderman observes anthropomorphic representations and concepts are rejected by users since they want to feel in control and do their own tasks [Shneiderman 95]. One study is not the final word, and in addition, we still don't know what features of the personified interfaces contribute to this rejection.

I believe there is a strong need for a theoretical study of visual representations for agents, which would result in guidelines for designing agent-based interfaces. In particular, this research is necessary before agents soon flood the world. I explore in this thesis the pros and cons of software agents with faces. Then I will discuss the effects of having faces and facial expressions in the interface on the basis of the results obtained by a series of experiments to compare subjects' responses to and evaluation of different faces and facial expressions of entertaining agents.

Contributions of this Research

The important findings of this research are as follows:

- Personified interfaces help users engage in a task, and are well suited for an entertainment domain.
- People's impressions of a face in a task are different from ones of the face in isolation. Perceived intelligence of a face is determined not by the agent's appearance but by its competence.
- The experimental results obtained demonstrate that there is a dichotomy between user groups which have opposite opinions about personification. Thus, agent-based interfaces should be flexible to support the diversity of users' preferences and nature of tasks.

Overview of This Document

Chapter 2 highlights related research on personified interfaces over the last decade, including Apple Knowledge Navigator [Laurel 90], Apple Guides [Oren 90], Newt [Sheth 94], and Maxims [Lashkari 94].

Chapter 3 describes the system design, user interface design, and character design of the poker game used for the experiments. Character design is an important step for creating appropriate representation for the experiments.

Chapter 4 describes the design of five experiments conducted on the World Wide Web. This chapter includes lessons learned from the test experiment, description about the procedure, the conditions, and the questionnaire for the web-based experiments.

Chapter 5 is the heart of this thesis. It describes the hypotheses of each experiment and discusses the results of the pilot experiment and the five web-based experiments--Face vs. NoFace, Male face vs. Female face, Human face vs. Animal face, Simple face vs. Caricature face vs. Realistic face, and three modes of Expressiveness.

Chapter 6 summarizes the findings of this research and discusses the future direction of studies on personified interfaces.