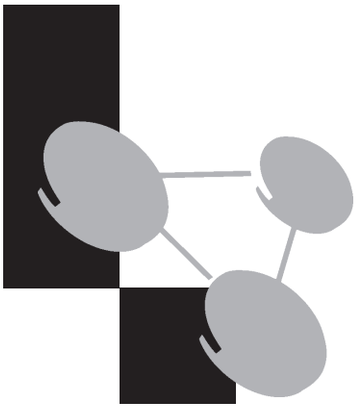

Virtual Reality (1999) 3:1-15

Spoken Language and Multimodal Applications for Electronic Realities

A. Cheyer, L. Julia

Computer Human Interaction Center (CHIC!), SRI International, Menlo Park, USA

Abstract: We use the term 'electronic reality' (ER) to encompass a broad class of concepts that mix real-world metaphors and computer interfaces. In our definition, 'electronic reality' includes notions such as virtual reality, augmented reality, computer interactions with physical devices, interfaces that enhance 2D media such as paper or maps, and social interfaces where computer avatars engage humans in various forms of dialogue. One reason for bringing real-world metaphors to computer interfaces is that people already know how to navigate and interact with the world around them. Every day, people interact with each other, with pets, and sometimes with physical objects by using a combination of expressive *modalities*, such as spoken words, tone of voice, pointing and gesturing, facial expressions, and body language. In contrast, when people typically interact with computers or appliances, interactions are unimodal, with a single method of communication such as the click of a mouse or a set of keystrokes serving to express intent. In this article, we describe our efforts to apply multimodal and spoken language interfaces to a number of ER applications, with the goal of creating an even more 'realistic' or natural experience for the end user.

Keywords:

Introduction

The primary objective in user interface (UI) design is to create a means for interacting with a computer or device that provides an intuitive yet efficient way for a person to accomplish a set of tasks. One way to achieve this goal is to model the UI after an environment in which the user is already an expert – the real world. For example, the popular desktop metaphor provides a loose conceptual framework based on the idea of moving papers across a desk.

Several UI approaches attempt a high-fidelity mixture of the real-world and computer interfaces. We refer to these fields of research as investigating 'electronic reality' (ER), and use the term to include notions such as:

- *Virtual reality (VR)*, where a 3D simulation creates an illusion for the user of the real world. Since the simulation is computer generated, elements in the world can be selectively enhanced in ways that physical law might prohibit.
- *Augmented reality (AR)*, where a computer overlays information and images on top of the user's real-world view. Both real and artificial objects are simultaneously viewable and able to be manipulated.

- *Interactive 2D Interfaces:* certain 2D displays, such as paper or whiteboards, can be enhanced to create interactive surfaces where the results are formed by cooperation between human and machine.
- *Social interfaces*, where computer avatars can participate in dialogues or interactions, perhaps operating in different roles than the usual command-and-control, master-slave relationship of typical human-computer interactions. Examples of such roles a computer avatar might assume include guide, critic, and instructor.

In the real world, people interact with each other, with pets, and sometimes with physical objects through a combination of expressive *modalities*, such as spoken words, tone of voice, pointing and gesturing, facial expressions, and body language. In contrast, when people interact with computers or appliances, interactions are typically unimodal, with a single method of communication such as the click of a mouse or a set of keystrokes serving to express intent. Our position is that spoken language and multimodal interactions should be an essential part of any ER system because of the familiarity and efficiency they bring.

In this article, we describe our efforts to apply multimodal and spoken language interfaces to a number of ER applications with the goal of creating an even more 'realistic' or natural experience for the end user. We begin with a brief overview surveying the component technologies used for constructing spoken language, natural language, and multimodal interfaces. We then present ER applications that have been further augmented with spoken language and multimodal interfaces. Finally, we conclude with a brief discussion of what we have learned and future directions.

Component Technologies

Speech Recognition

Automated speech recognition (ASR) has made significant strides in the past few years, in part because of hardware advances that have brought the required computational power onto a large number of desktop machines. It is now possible to purchase large-vocabulary continuous dictation products with word-accuracy rates as high as 95% for less than fifty dollars. Commercial implementations include IBM's ViaVoice[1,2], Dragon's Naturally Speaking[3], and Lernout & Hauspie's Voice Xpress[4].

Although dictation products have improved, many speech-enabled applications, both commercial products and research prototypes, do not use large-vocabulary dictation systems as their front ends. Systems targeting a particular domain often find that recognition accuracy can be improved by tailoring the speech models to the application-specific tasks. Alternatives for doing so include grammar-based approaches, where an engineer explicitly defines the 'tree' of possible utterances, and statistical models computed across a specialised corpus of collected speech in the target domain. Nuance Communications[5], one of the leaders in speech-enabled telephony applications, uses grammars to empower telephone-based applications for stock quote retrieval or accessing email and calendar information. IBM's ViaVoice developer tools include compilers and APIs for using speech grammars. Also offered is the 'ViaVoice Topic Factory', which enables developers to hone a larger dictation model down to their particular domain.

Natural Language Interpretation

Recognising spoken words is only the first part of the problem – once they are recognised, they must be interpreted. Natural language (NL) engines provide the means to map an input request in English or another language into some internal representation, often called a logical form, that can be processed by the application. NL components come in varying forms:

- If the speech models are grammar based, with all possible utterances explicitly coded in a tree, a simple way of processing the returned words is through coding *annotations* or return values directly into the grammar for each sentence. Processing the resulting tokens or slot-values is much simpler than handling all combinations of possible utterances.

[1]IBM Via Voice: http://www.software.ibm.com/speech/

[2]All product or company names mentioned in this document are the trademarks of their respective holders.

[3]Dragon Systems: http://www.dragonsys.com/

[4]Lernout & Hauspie's Voice Xpress: http://www/lhs.com/voicexpress/

[5]Nuance Communications: http://www.nuance.com

- Frame-based NL systems can be defined for a specific domain, and patterns used to pull important values from the recognised text. For example, in an air-travel domain, detecting '…to Boston…' might be sufficient to fill the *destination* slot without having to comprehend fully every word in the input. Frame-based approaches can be used effectively with dictation engines, which can produce agrammatical results.
- NL parsing components attempt to use rules of language grammar to decompose a sentence into its primary parts and to produce a context-independent structured representation of the utterance's meaning. Parsers may be top-down, with grammar rules looking for words, or bottom-up, mapping words into progressively bigger fragments by using the rules. An example of a sophisticated NL system is Gemini, a bottom-up parser that simultaneously interleaves syntactic and semantic features by using 'unification grammars'. Gemini includes optional modules for handling repairs and error corrections, and can robustly handle agrammatic sentences. [1].

Integration Frameworks

In constructing a system that will involve multiple technologies such as speech or handwriting recognition engines, natural language parsers, and knowledge sources for encoding multimodal fusion strategies, an integration framework can allow rapid development and experimentation through plug-and-play swapping of components. When evaluating an integration framework, we feel that it is important to look at whether the infrastructure supports integrating components written in different languages – many NL systems are written in languages such as Lisp or Prolog, while speech recognition engines typically offer C-based interfaces. We must think also about where interactions among components are coded, and how easy this procedure is to upgrade if new components are added to or removed from the system.

In the commercial world, object-oriented or distributed object-oriented approaches such as the Object Management Group's CORBA[6], Microsoft's DCOM[7], or Sun's RMI[8] enable the construction of complex, multicomponent systems. CORBA is language and platform independent, DCOM is language independent but is primarily intended for use with Microsoft Windows platforms, and RMI offers platform- independent but language-specific computing in JAVA. We shall now look briefly at two frameworks being used by the research community for constructing spoken-language and multimodal applications. Although these frameworks do not currently provide the scalability and robustness of the commercial products, they do offer more support for encoding flexible interactions among heterogeneous distributed components.

The DARPA Communicator program is a government-sponsored research effort focusing on next-generation conversational interfaces to distributed information. Although focusing initially on telephone-based dialogue, the goal is to support the creation of speech-enabled interfaces that scale gracefully across modalities, from speech-only to interfaces that include graphics, maps, pointing, and gesture.

As an integration framework, the Communicator program has selected an architecture based on MIT's Galaxy framework [2] as the community-wide standard. This architecture provides a scriptable 'hub' that controls all interactions among a population of server programs providing functions such as speech recognition and text to speech. The advantages of this approach are that individual recognition servers can be replaced very easily in a plug-and-play manner, and that the servers are easily reusable in other domains. Since all of the interaction specifications are located in the hub scripts, porting to a new domain involves changing code in only one place instead of many. However, one major disadvantage is that server components are stateless, and any state they need must come in advance at the time of the request for service. Servers are not able to initiate a request of the hub or of other servers, which at times creates difficulty in representing certain interactions.

In many of the applications we describe in this article, we chose to use the Open Agent Architecture™ (OAA[9]) as our implementation framework [3]. The OAA is a general-purpose infrastructure for constructing systems composed of multiple software components written in different programming languages and distributed across multiple platforms [4]. Similar in spirit to distributed object frameworks

[6]CORBA: Common Object Request Broker Architecture.

[7]DCOM: Distributed Component Object Model.

[8]RMI: Remote Method Invocation.

[9]More information can be found on the OAA homepage at http://www.ai.sri.com/~oaa

such as CORBA or DCOM, OAA provides support for describing more flexible and adaptable interactions than the tightly bound method calls provided by these architectures. In addition, OAA's facilitation-based approach provides numerous services suitable for developing multimodal applications, including the following:

- Agents communicate using a logic-based tasking language called ICL. Several agent-enabled systems exist that can translate from English to ICL and back to English, enabling users to interact closely with agents in a natural way.
- The infrastructure, through Facilitator agents, supports conflict management, competitive and cooperative parallelism, failure conditions across multiple agents, and so forth.
- OAA has built-in support for developing collaborative applications where multiple humans and agents share the same workspace.

OAA has been used to implement more than 30 applications in various domains, many of them multimodal in nature [5]. OAA has also been used by organisations outside SRI. Examples include Oregon Graduate Institute's QuickSet system [6] and Ecole Polytechnique Fédérale de Lausanne's telepresent surgical simulations [7].

Multimodal Applications of Electronic Realities

Researchers at SRI and elsewhere [8, 9] have applied spoken language and multimodal interfaces to various types of electronic realities.

Augmenting Drawing Surfaces

Our first work with extended input peripherals and alternative interface metaphors focused on adapting a user's interaction with a pen and piece of paper to the electronic realm. In the TAPAGE/DERAPAGE applications (Fig. 1), a user imagines a complex nested table or flowchart, draws a rough freehand sketch of the concept, and then engages in an interactive dialogue with the system until the desired product is realised [10]. The system does its best to interpret the intention of the user, rendering a 'clean' version of the drawing with lines straightened, columns centered, and so forth. The user can incrementally edit the figures by using natural combinations of both pen and speech, crossing out an undesirable line, drawing in new additions, and repositioning lines or objects with commands such as 'put this over here'. During multimodal utterances, input may be entered simultaneously or in any sequential order. In these applications, we use a frame-based model called VO*V*, whose three slots represent the main verb, one or more objects to which the verb applies, and additional variables (attributes) necessary to complete the command.

The primary goal of TAPAGE/DERAPAGE is to capture the nature of a pen-and-paper experience, while enhancing the paper's role to become a partner in the process, capable of following high-level instruction and being active in the construction of the document. During the evaluation phase of the project, both novice and expert users were comfortable using familiar pen and voice modalities, and were able to accomplish design tasks much more efficiently than with a conventional keyboard/mouse interface for the same task (e.g. Microsoft Excel).

There is a significant body of work related to augmented drawing interfaces. Similar in functionality to DERAPAGE but operating in an offline mode rather than incrementally, one application is capable of automatically processing hand-drawn flow chart diagrams [11]. A number of augmented whiteboard projects provide other examples of computer-augmented drawing surfaces, although they do not contain the speech component of TAPAGE/DERAPAGE. In FlatLand, an enhanced whiteboard provides space and history management, as well as supplying several automated 'behaviours' such as list manipulation and route-drawing cleanup [12]. In the i-Land system, a computer-augmented whiteboard allows collaborative development and sharing of documents in conjunction with other devices such as an 'InteracTable' and wireless mobile computers [13]. In the DigitalDesk project [14], documents placed or projected onto a desk can be manipulated with pens and with bare fingers. Instead of making the workstation more like a desk (c.f. Windows 'desktop' metaphor), the goal is to make the desktop more like a workstation. This interesting AR project actually gives electronic capabilities to real objects and, in particular, to paper.

Multimodal Maps

Our next ER project, a variant of the 'smart paper' domain, focused on maps, where the goal is to manipulate and reason about information of a

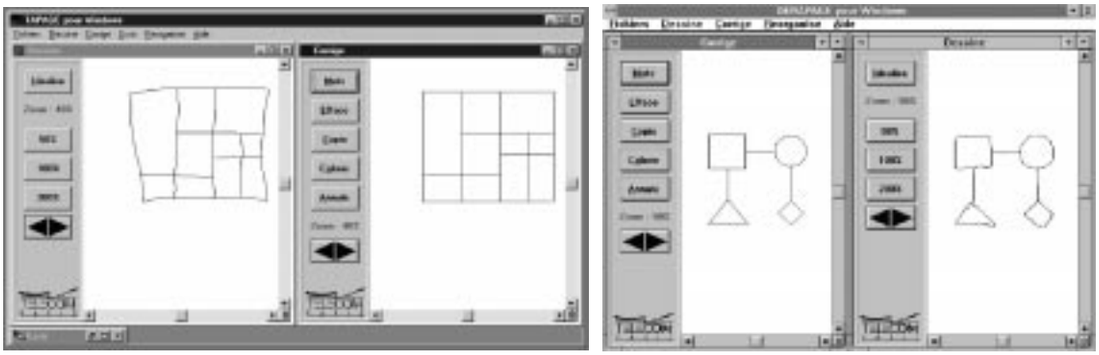

Fig. 1. TAPAGE and DERAPAGE: interactive paper using pen and voice.

geographic nature (Fig. 2). Inspired by a simulation experiment [15], we developed a working prototype of a travel planning application, where users could draw, write, and speak to the map to call up information about hotels, restaurants, and tourist sites [16]. Whereas our previous efforts were more concerned with algorithms for robustly handling freehand drawings and gestures, our multimodal map application demands more from the NL component and speech components. A typical spoken interaction might be *'Find all French restaurants within a mile of this hotel'* + *<draw arrow toward a hotel>*. The NL component can also handle abbreviated utterances that often arise from handwriting, such as 'Ggate pk?'

The primary research challenges involved in constructing such a system are in how to develop a multimodal engine capable of blending incoming modalities in a synergistic fashion, able to resolve the numerous ambiguities that arise at many levels of processing. One problem of particular interest was that of reference resolution (anaphora). For example, given the utterance 'Show photo of *the* hotel', several distinct computational processes may compete to provide information: a natural language agent may volunteer the last hotel talked about;

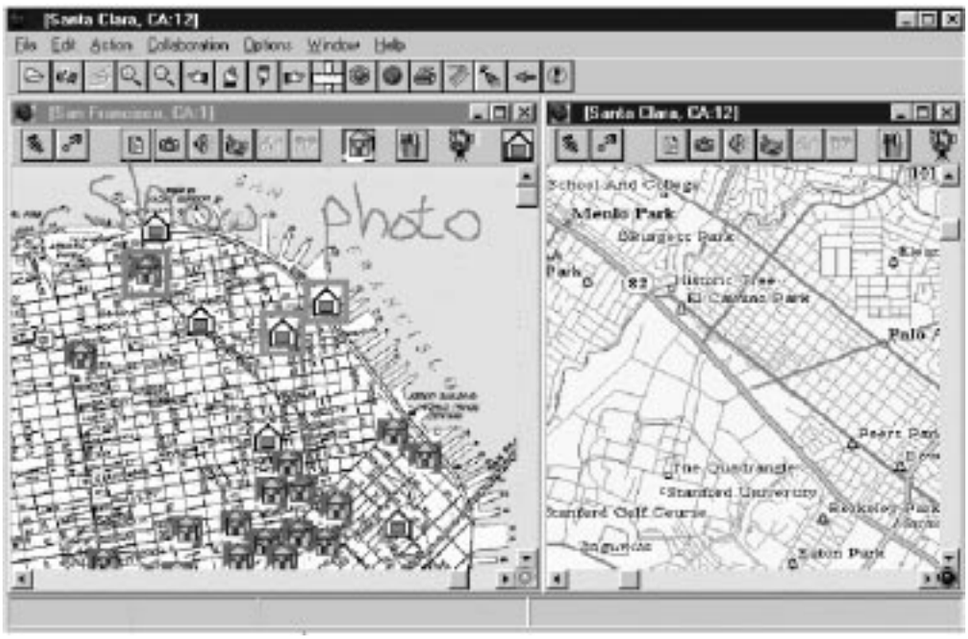

Fig. 2. Multimodal maps.

the map interface might indicate that the user is looking at only one hotel; and, a few seconds later, a gesture recognition process might determine that the user has drawn an arrow or circled a hotel. If the request is 'Show photo of *the* hotel on Main Street', a database containing the addresses of hotels must cooperate with the other competing knowledge sources. In certain situations, a request may be truly ambiguous: in this case, the user can be brought into the dialogue to indicate the specific hotel by writing its name, speaking, or selecting the hotel by pointing or drawing. To implement the reference resolution strategies and multimodal fusion algorithms, our approach makes use of OAA's facilitation services for coordinating parallel processes, enabling an approach that is extensible and adaptable to user preferences.

Maps, being very graphical in nature, have been the subject of several multimodal-related research projects. The effort most similar to our work is Oregon Graduate Institute's Quickset prototype [6]. QuickSet, a pen- and voice-enabled mapping system for a military domain, is also implemented using SRI's OAA, so it shares many of the properties of our approach. However, instead of using a distributed reasoning approach to multimodal fusion, QuickSet's interpretation is based on 'semantic unification'. The advantage of this approach is that robust error correction can be obtained using a combination of n-best recognition results from multiple recognisers. However, the associative memory used by the approach is suitable only for a limited set of simple NL commands such as 'scroll map' and does not scale well to the more complex queries required by our travel planning domain.

In the CARTOON (CARTography and cOOperatioN between modalities) system, the user can speak and point to a map (with a mouse) to produce queries such as 'I want to go from here to here' or 'Where is the police station' [17]. Multimodal interpretation is handled by 'guided propagation networks' composed of simple processes using event detectors and fusion nodes. Activation across various nodes modulated by temporal proximity contributes to a highest interpretation. One of the advantages of this approach is that some of the properties in the network can be deduced through learning algorithms. However, as with QuickSet's semantic unification, no true parse tree is constructed for natural language, so complex queries involving relationships and attributes are probably beyond the scope of this system.

Other research on multimodal map systems can be found in [18, 19].

Multimodal interfaces for synchronised 2D and 3D spaces

Through the previous experiments and prototypes, we were able to develop some sense of multimodal interactions that could be used to enhance 2D

Fig. 3. Multimodal interactions in synchronised 2D and 3D maps.

drawing surfaces. However, with 3D becoming more prominent in user interfaces [20], we were thus curious about whether the same input techniques (i.e. drawing, writing, speaking) would be effective for 3D situations.

To create an environment in which to pursue this investigation, we began by augmenting our 2D map by a 3D VR model of the world (Fig. 3). In the resulting system, a user can choose to work in either a 2D window (map – bird's eye view) or a 3D window, and the two are kept synchronised, with viewports and object information icons updated simultaneously in both.

Although many commands remain primarily the same in both 2D and 3D worlds (e.g. 'Bring me to the Hilton'), it is unclear how best to interpret both pen gestures and speech utterances for 3D. For instance, does an arrow to the left indicate that the user wants to turn toward the left, keeping the same position, or rather pan her position toward the left, keeping the same orientation? What does the spoken reference 'up' mean in the context of complex 3D terrain? Although clearly a 2D paper metaphor does not transparently map onto a 3D environment, we have begun conducting more detailed experiments focusing on pen–voice interactions for 3D models, specifically looking at:

- Deictic and gestural reference to features of the terrain: how do people refer to and distinguish between features of a terrain model by using words and gesture?
- Discourse structure: how does the structure of the interaction enable more economical communication, and how can a computer system utilise this structure in interpreting spoken and gestural input? How is the discourse structured by the structure of the terrain model and of the task or operation being executed in the terrain?
- Spatial language: how does language carve up space, and what is its relation to more geometric representations of space used in terrain models?

Another example of a multimodal map application that integrates a synchronised 2D and 3D view is the CommandTalk/CommandVu system developed as part of NRaD's LeatherNet system [21]. CommandTalk is a spoken language interface to a military map simulator. Adding speech to the application served several purposes: first, interaction with the system was greatly improved in terms of productivity – instead of accessing large menus with literally hundreds of choices, a user can simply describe the desired function by using voice. In addition, using speech to control simulated military forces provides an effective tool for training personnel in the proper techniques for giving commands over the radio. In CommandTalk, spoken language and pen or mouse gestures can be used to:

- Create forces and control measures – 'Create an M1 platoon here' + deictic gesture, or a crossout gesture to delete a unit.
- Assign missions to forces – 'First platoon, on my command, advance in a column to Checkpoint 1, using this path' + drawing of the path.
- Modify missions during execution – 'Change formation to echelon right'.

Fig. 4. CommandVu: VR views synchronised with CommandTalk map simulator.

- Control system functions, such as the display 'Centre on objective Alpha', or right arrow to scroll the map.

CommandVu is a 3D view of the CommandTalk simulation (Fig. 4). An operator may use CommandTalk to set up a simulation exercise and then, as the battle unfolds, monitor the experience by using an enclosed VR display with 3D audio explosions. CommandTalk and CommandVu are multiuser sytems so multiple operators can play different roles within the simulation environment.

Multiuser Collaboration

Multiuser collaborative environments are becoming increasingly popular, particularly in 3D virtual domains. One example of such a system is the Distributed Interactive Virtual Environment (DIVE), a multiuser VR system where distributed participants navigate in 3D space and interact with other users and applications [22]. First appearing in 1991, DIVE is freely available for noncommercial use, supports VRML and most 3D formats, and is especially optimised for efficient network use. Another multiuser framework for virtual interactions is SHAVE (SHAred Virtual Environment) [23]. SHAVE features dynamic connection and disconnection with the system, user-definable 3D avatars, and automated agents to handle behaviors in the system. SHAVE has been designed to handle a large number of distributed users.

While SHAVE and DIVE enable distributed collaboration among users in a 3D world, neither focuses on multimodal interactions. However, the MASSIVE collaborative virtual environment looks at the interesting problem of using 3D space to intelligently filter multimodal messages to the appropriate human and inanimate participants [25]. The problem that MASSIVE's developers are trying to solve is not only how to add the intelligence to objects that would enable them to interpret multimodal streams, but how they could understand when utterances they 'overhear' are intended for others. By applying spatial awareness techniques to virtual objects, MASSIVE provides nonhuman objects a better sense of social awareness.

In our work, we are beginning to investigate how speech recognition can be used in 3D multiplayer

Fig. 5. Speech for an immersive virtual game: 'Follow that dolphin'.

games (Fig. 5). We believe that this form of interaction will have an impact in the 3D gaming arena, and are planning to investigate these possibilities more closely.

We have also carried out some exploration of spaces where multiple humans and autonomous agents can interact in a shared environment. For example, in the multimodal map application discussed earlier, a human user can draw, write or speak to interact with a community of distributed information agents, and agents themselves compete and cooperate among themselves to resolve tasks for the user. As the OAA framework in which the multimodal map system is implemented provides some data-replication primitives, it is an easy matter to design the application such that it is multiuser capable. Any workspace window can be shared with other members of a workgroup and, as interactions occur in the window, the state is replicated so that all users can view the changes. This creates an interesting dynamic: certain pen interactions may be acted upon by an agent (for instance, a drawn line might provoke an agent to calculate the distance along the path), whereas other pen input is meant for human consumption (a path drawn as an illustration of some concept). We are only beginning to explore the interesting issues that can arise, for example:

- If an agent can answer a question quickly, should the agent always respond immediately, or is it more polite to give another human a chance to reply?
- If an agent is very slow in answering a question or in finding some information related to the context, should the agent interrupt the current discussion? Perhaps the topic has since changed and the information is no longer relevant.

To explore answers to these questions, our approach is to use a series of user experiments that let us quickly evaluate possible solutions in both real and simulated situations.

User Experiments: the WOZZOW Simulation

As part of our application development, we have found it essential to integrate user feedback, both during the design phase when we are imagining what functionality the system should provide and how the system will behave, and after the prototype is functional to evaluate where our implementation and algorithms succeed or fail. For the design phase, Wizard of Oz (WOZ) simulations have proven an effective technique for discovering how users would interact with systems that are beyond the current state of the art [15]. In a WOZ setup, users are brought in to interact with a new system and their interactions recorded to reveal what they do. Although they are led to believe the system they are working with is fully automated and functional, a hidden 'wizard' monitors their input and uses a configuration panel to remotely control their application. The advantage of such a system is that the user is not constrained by the limitations and assumptions made by an implemented system.

We describe a novel extension to the WOZ methodology, that we call a WOZZOW simulation [4]. A WOZZOW simulation is a technique for simultaneously running two experiments in one: the WOZ half operates like a standard Wizard Of Oz simulation to collect data from naïve users in the manner described above. The ZOW half collects end-user data from an expert user, evaluating how well our best fully functional prototype system is working. The simulation technique, which makes use of the collaboration and synchronisation capabilities of OAA, works as follows (using multimodal map as an example):

- Instead of constructing a specialised simulation environment whose sole purpose is to collect data from users, we run a real, working OAA application in multiuser collaboration mode so the displays are synchronised. One display is configured in a minimalist way, with no scrollbars, toolbars, or buttons, to allow only pen and voice input; the other is presented with all system dialog boxes and GUI controls visible (Fig. 6).
- An uninitiated user (the 'subject') is told to write, draw, or speak to the system to accomplish a complex task such as planning a weekend in Toronto (Fig. 6). In a second room is hidden our wizard, an experienced user of the application, whose role is to perform the actions requested by the subject as quickly as possible, using any combination of pen, voice, or GUI controls. In this way, the subject is led to believe that the system is interpreting his input. In the case of the wizard, the system really is processing her multimodal requests.
- In a single experiment, we simultaneously collect data input from both an unconstrained new user (unknowingly) operating a simulated system – providing answers about how pen and

Fig. 6. A WOZZOW simulation – synchronised interactions between two users. Left : what the WOZ subject sees: an unadorned map encouraging pen and voice input; Right: the ZOW expert chooses either multimodal commands or standard GUI controls.

voice are combined in the most natural way possible – and from an expert user (under duress) making full use of our best automated system. In analysing the wizard's interactions, we can learn how well the real system performs, and investigate the roles of a standard GUI (e.g. buttons, scrollbars) relative to a multimodal interface.

A WOZZOW simulation provides numerous advantages over a standard WOZ simulation:

- There is a very low cost in turning an OAA application into a WOZZOW simulation thanks to OAA's built-in collaboration, logging and playback facilities.
- Resulting improvements to the end-user system garnered from the experiments are *quantifiable*. Groups of subject input data can be run over the real system before and after findings are incorporated (e.g. enhancing speech grammars, fusion algorithms), and the rate of success can be measured.
- An application develops in an incremental style, where the performance of the real system is tested even as the simulation side of the

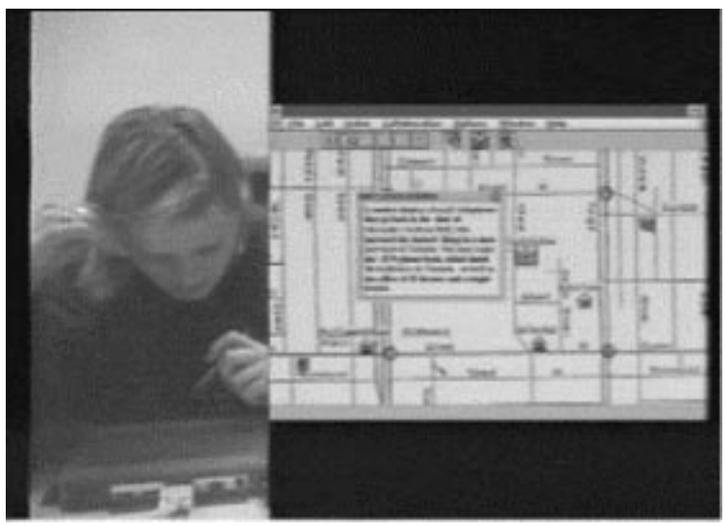

Fig. 7. Recorded video of WOZZOW subject working with the multimodal map**.**

experiment provides information about future enhancements.

In [25, 26], we provide initial results of experiments using this approach for the multimodal map application (Fig. 7).

Augmenting the Real World with a Virtual World

Although pen-and-voice input seems to be a potentially promising device for interacting with 3D environments, we are looking for solutions that provide less intrusive and even more natural interactions. Sensors are now becoming available that allow computer systems to monitor a user's position, orientation, actions and views, and construct a model of the user's experience. Access to such a model will enable computer programs to proactively and continually look to enhance the user's real-world perceptions, without specific intervention from the user. Display devices allow the computer to overlay additional information directly over the user's view of the real world. This concept is popularly known as 'augmented reality' (AR).

To facilitate exploration of the augmented reality paradigm, we have been constructing an AR application framework, called the Multimodal Augmented Tutoring Environment (MATE). In this framework, multiple processes for providing sensor readings, modality recognition, fusion strategies and viewer displays, and information sources can be quickly integrated into a single flexible application. Our first AR prototype 'Travel MATE' (Fig. 8) makes use of many of the technologies developed in our 2D and 3D tourist applications, but adds GPS and a compass sensor[10]. As a user walks or drives around San Francisco, a small laptop computer or PDA simultaneously displays a 3D model of what he is seeing in the real world, automatically updated based on his position and orientation. If a user wants to know what a particular building in the distance is, she can look at the display where objects in view are labelled. More detailed multimedia information about these objects can be retrieved on request. We are also working on an 'Office MATE' prototype to investigate how AR and situated awareness could enhance the workplace.

Projects similar to Travel MATE include NaviCam [27] and Ubiquitous Talker [28]. The NaviCam system is made up of a small handheld video camera attached to an LCD screen. The metaphor for NaviCam is of a magnifying glass with which a user looks at the world. When NaviCam detects particular

[10]More information about the Travel and Office MATE projects can be found at http://www.chic.sri.com/projects/MATE.html

Fig. 8. Travel MATE, augmenting tourist experiences.

Fig. 9. InfoWiz, SRI's interactive kiosk.

objects through the use of a bar-code reader, the system augments the image shown on the display with additional information, magnifying the image not visually but intentionally. Ubiquitous Talker is an extension of NaviCam that includes a spoken language system. As users examine objects through the camera, they can ask questions of the objects, which respond through textual displays. NaviCam and Ubiquitous Talker differ from Travel MATE and Office MATE in the clever use of an integrated display device and camera, and in their sensing mechanism (bar-code reader instead of GPS+compass). The MATE projects add multimodal interactions to the spoken language interface through the use of a touch screen. Another style of interaction between machine and virtual world is described the InfoWiz kiosk application (Fig. 9). The project is centered around the idea of putting an interactive kiosk into the lobby of SRI [29]. Instead of presenting a touch screen or mouse to navigate through the information, all interactions with the kiosk occur through spoken requests to an animated cartoon character known as the InfoWiz, issued into a telephone (a real-world, familiar interface). In looking for solutions, this work has been influenced by [30, 31, 32, 33, 34].

Interacting with Physical Agents

SRI was perhaps the birthplace of mobile robots, constructing Shakey in 1966. Robots still roam the halls and, what's more, they are starting to work together. In 1996, SRI won the 'Office Navigation' event at the AAAI Robot Competition, using a team of cooperating robots [35]. Recently, we have been working on constructing a wearable user interface that will enable a human to work with the robots as part of the team [36].

Figure 10 depicts a concept drawing for such a device. Imagine that a SWAT team needs to respond to a terrorist takeover of a building. Since this is a dangerous mission, the team brings several mobile autonomous robots equipped with adjustable video cameras, audio, and other sensors. An arm-mounted device provides a configurable display for controlling and tasking the robots and their sensors.

Although our current prototype runs on a laptop equipped with a touch screen instead of a wearable computer, the system is able to provide multimodal interactions to a team of wireless robots. In addition to directing robots through a multimodal map-style interface (e.g. 'You are here facing this direction. Go pick this up.'), and controlling and annotating robot's video input (e.g. 'Zoom in on this. Grab this region for the report.'), pen and voice are used in a cooperative map-building task. An operator with a general idea of a floor space layout can sketch a rough map and indicate constraints on individual entities. The result is cleaned up (using algorithms from TAPAGE and DERAPAGE) and sent to the robots, which attempt to match their local sensors to the global map, updating information as they go, as shown in Fig. 11. Clarification dialogues may be required between human and mobile machines: a robot may buzz the user and ask if the view from

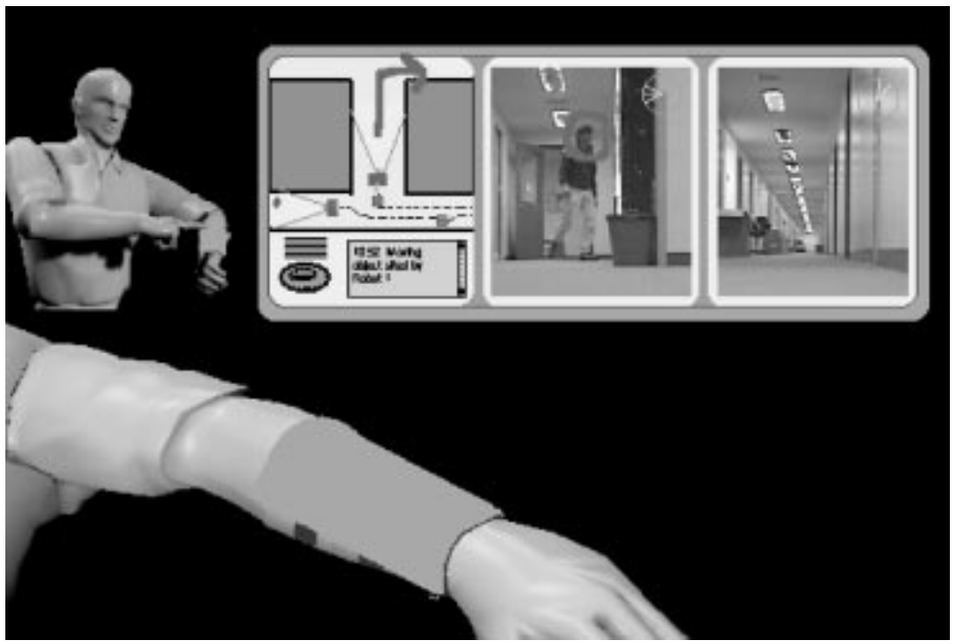

Fig. 10. Concept for wearable robot tasking device.

its video camera corresponds with what the user was expecting.

Related work conducted at the Pennsylvania State University Virtual Tools and Robotics Group attempts to provide ways for robot operators to direct robots with gestures and natural language [37]. Sample comands might include 'put that there', 'cut there', 'polish there', or other actions where a human can supply positional information to aid a robot's task. Application domains include flexible manufacturing, hazardous waste remediation, and space telerobotics.

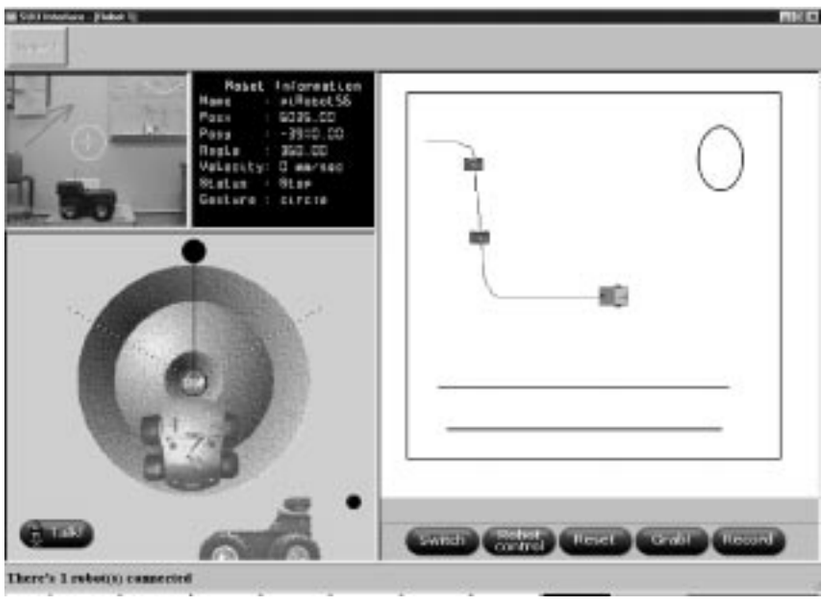

Fig. 11. Current prototype for tasking a team of robots and their sensors.

Conclusions

The metaphors we use today to interact with computers were developed primarily in the 1960s and 1970s by researchers from SRI and Xerox. As computers, sensors, bandwidth, display capabilities, and software techniques continue to improve at incredible rates, providing computational power only dreamed of during the 1960s and 1970s, opportunities are emerging to transform the paradigms used in human-computer interaction. One of the most promising areas of research for creating new forms of human computer interaction belongs to the family of interfaces we are calling 'electronic realities'. It is our position that these interfaces will be further improved by incorporating spoken language and multimodal interaction styles.

In this article, we have discussed some of our research efforts exploring the combination of ER applications and multimodal interfaces and placed them in context of related work. Applications were presented in the domains of 'smart paper', multimodal maps, synchronised 2D and 3D displays, multiuser collaborative environments, embodied dialogue systems, and multirobot control. We also described our approach for simultaneously designing, implementing, and evaluating multimodal applications by using an incremental process and the WOZZOW simulation methodology.

Our current work as part of the SRI's Computer Human Interaction Center (CHIC!) will continue to perform user evaluations in multimodal systems, with a particular emphasis on exploring language and reference for 3D worlds. New prototype applications are under way in the domains of smart spaces, augmented meetings, and applications to improve human-device interactions in the home.

References

1. Dowding J, Gawron JM, Appelt D, Bear J, Cherny L, Moore R and Moran D. Gemini: a natural language system for spoken-language understanding. In: Proceedings of 31st Annual Meeting of the Association for Computational Linguistics, Columbus, OH, 1993
2. Goddeau, D, Brill, E, Glass, E, Pao, C, Phillips, M, Polifroni, J, Sene, S. and Zue, V. GALAXY: a human-language interface to on-line travel information. In: Proceedings ICSLP, 1994
3. Martin, D, Cheyer, A, Moran, D. The open agent architecture: a framework for building distributed software systems. Applied Artificial Intelligence, 1999; 13: 1–2.
4. Cheyer, A, Julia, J, Martin, JC. A unified framework for constructing multimodal experiments and applications. In: Proceedings CMC'98, 1998
5. Moran D, Cheyer A, Julia L, Martin D, Park S. Multimodal user interfaces in the open agent architecture. Journal of Knowledge-Based SYSTEMS 1998; #10, 295–303
6. Cohen, PR, Johnston, M, McGee, D, Smith, I, Oviatt, S, Pittman, J, Chen, L, Clow, J. QuickSet: multimodal interaction for simulation set-up and control. In: Proceedings of the Fifth Applied Natural Language Processing Meeting, Association for Computational Linguistics, Washington, DC, 1997
7. Baur, C, Guzzoni, D, Georg, O. A virtual reality and force feedback based endoscopic surgery simulator. In: Proceedings MMVR'98, San Diego, CA, January 1998, 110–116
8. Bolt R. Put that there: voice and gesture at the graphics interface. Computer Graphics 1981; 14(3), 262–270
9. Cohen PR. Synergistic use of direct manipulation and natural language. In: Proceedings CHI'89, New York, NY, 1989; 227–233
10. Julia, L, Faure, C. Pattern recognition and beautification for a pen based interface. In: Proceedings ICDAR'95, 1995
11. Yu, B. Automatic understanding of symbol-connected diagrams. In: Proceedings of the Third International Conference on Document Analysis and Recognition, 1995
12. Mynatt, E, Edwards, WK. Flatland: new dimensions in office whiteboards. In: Proceedings of CHI'99, Pittsburgh, PA, 1999; 346–353
13. Streitz, N, Geibler, J, Holmer, T, Konomi, S, Muller-Tomfelde, C, Pexroth, P, Seitz, P. i-Land: an interactive landscape for creativity and innovation. In: Proceedings of CHI'99, Pittsburgh, PA, 1999; 120–127
14. Wellner, P. Interacting with paper on the DigitalDesk. Communications of the ACM, 1993; 36(7): 87–96
15. Oviatt, S. Multimodal interfaces for dynamic interactive maps. In : Pr oceedings CHI'96, 1996
16. Cheyer, A, Julia, L. Multimodal maps: an agent-based approach. In: Multimodal human-computer communication, Lecture Notes in Artificial Intelligence #1374, Springer, 1998
17. Martin, JC. Towards «intelligent» cooperation between modalities. The example of a system enabling multimodal interaction with a map. In: Proceedings of the IJCAI-97 workshop on intelligent multimodal systems, Nagoya, 1977
18. Siroux, J, Guyomard, M, Multon, F, and Remondeau, C. Modeling and processing of the oral and tactile activities in the Georal tactile system. In: Proceedings CMC'95, 1995
19. Neal, JG, and Shapiro, SC. Intelligent multi-media interface technology. In: Intelligent user interfaces. Sullivan, JW, Tyler, SW, eds. Reading: Addison-Wesley, 1991; 11–43
20. Ark, W, Dryer, C, Selker, T, Zhai, S. Representation matters: the effect of 3D objects and a spatial metaphor in a graphical user interface. In: Proceedings of CHI'98, 1998
21. Julia, L, Cheyer, A, Dowding, J, Bratt, H, Gawron, JM, Bratt, E, Moore, R. How natural inputs aid interaction

in graphical simulations. In: Proceedings VSMM'98, Gifu, Japan, 1988; 466–468

22. Carlsson, C, Hafsand, O. Dive: A multi-user virtual reality system. In: Proceedings of the IEEE 1993 Virtual Reality Annual International Symposium, 1993; 394–401
23. Amselem, D. A window on shared virtual environments. Journal Presence: Teleoperators and Virtual Environments, 1995; 4(2)
24. Benford, SD, Greenhalgh, CM. A spatial approach to speech and gestural control in collaborative virtual environments. In: Proceedings of the Combined International Conference on Artificial Reality and Tele-Existence '95 and ACM Symposium on Virtual Reality Software and Technology '95 (ICAT/VRST'95), 1995; Makuhari: Chiba,
25. Martin, JC, Julia, L, Cheyer, A. A theoretical framework for multimodal user studies. In: Proceedings CMC'98, Tilburg, 1988
26. Kehler, A, Martin JC, Cheyer A, Julia L, Hobbs J, Bear J. On representing salience and reference in multimodal human-computer interaction. In: Proceedings AAAI'98 (Representations for Multi-Modal Human-Computer Interaction), Madison, 1988; 33–39
27. Rekimoto, J, Nagao, K. The world through the computer: computer augmented interaction with real world environments. In: Proceedings CHI'98, 1988
28. Nagao, K. Agent augmented reality: agents integrate the real world with cyberspace. In: Community computing: collaboration over global information networks. Ishida, T. ed. John Wiley & Sons Ltd., 1988
29. Cheyer, A, Julia, L. InfoWiz: an animated voice interactive information system. In: Proceedings Agents'99: Workshop on Conversational Agents and Natural Language, 1999
30. Andre, E, Rist, T, Muller, J. Guiding the user through dynamically generated hypermedia presentations with a life-like character. In: Proceedings of International Conference on Intelligent User Interfaces (IUI-98), 1988
31. Cassel, J, Bickmore, T, Billinghurst, M, Capbell, L, Change, K, Vilhjalmsson, H, Yan, H. Embodiment in conversational interfaces: Rea. In: Proceedings of CHI'99, Pittsburgh, PA, 1999; 520–527
32. Towns, S, Callaway, C, Voerman, J, Lester, J. Coherent gestures, locomotion and speech in life-like pedagogical agents. In: Proceedings of Intelligent User Interfaces (IUI-98), 1988; 13–20
33. Rousseau, D, Hayes-Roth, B. Personality in synthetic agents. In: Knowledge systems laboratory report No. KSL 96–21, Stanford University, 1996
34. Nagao, K, Takeuchi, A. Speech dialogue with facial displays: multimodal human-computer conversation. In: Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL-94), 1994; 102–109
35. Guzzoni D, Cheyer A, Julia L, Konolige K. Many robots make short work. Report of the SRI International mobile robot team at AAAI96. AI Magazine, Spring 1997, 55–64
36. Julia, L. Tasking robots through multimodal interfaces: the coach metaphor. In: Collective robotics, Lecture Notes in Artificial Intelligence #1456, Drogoul ed., Springer, 1988; 38–47
37. Cannon, DJ, Leifer, LJ. Point-and-direct robotics, In: Proceedings of International Conference on Intelligent Teleoperation, 1991; 95–106

Copy and offprint requests to: *Computer Human Interaction Center (CHIC!), SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025, USA. Email adam.cheyer@sri.com or luc.julia@sri.com http://www.chic.sri.com*