Editors: Lawrence Rosenblum and Michael Macedonia

Multimodal Interaction for 2D and 3D Environments

Philip Cohen, David McGee, Sharon Oviatt, Lizhong Wu, and Joshua Clow

Oregon Graduate Institute of Science and Technology

Robert King, Simon Julier, and Lawrence Rosenblum

Naval Research Laboratory

The allure of immersive technologies is undeniable. Unfortunately, the user's ability to interact with these environments lags behind the impressive visuals. In particular, it's difficult to navigate in unknown visual landscapes, find entities, access information, and select entities using 6 degrees-of-freedom (DOF) devices. We believe multimodal interaction—specifically speech and gesture—will make a major difference in the usability of such environments.

Multimodal interaction

Voice and gesture can complement an immersive environment. Consider a handheld 2D interactive map-based system used in a 3D environment to provide a "birds-eye" view of the scene. Such a map system enables a user to discover information out of the field of view (FOV) and to interact with the 2D representation of the entities in the 3D scene. Researchers have investigated direct manipulation interaction when the 2D map is displayed in the 3D scene (a configuration appropriate for head-mounted displays) or on a portable device. Voice/gesture-based interaction with 2D maps demonstrably offers significant speed, robustness, and user preference advantages over graphical user interfaces (GUIs) and speech-only interfaces. With voice, the user can issue commands for simulated flight or "teleportation" to arbitrary locations without having to navigate in the virtual environment, can find entities in a large 3D space, and can access collateral information by asking arbitrary questions. With 2D gesture, the user can easily select 2D representations of objects in the scene, draw entities to appear in the scene, and describe simulated flight paths.

1 QuickSet running on a tablet PC.

Direct voice and gesture interaction with the 3D scene offer benefits analogous to those discussed above for 2D visualizations. Furthermore, multimodal 3D interaction may provide a more robust interface than can speech or gesture alone. Here we illustrate reasons why we believe a multimodal approach has promise for 3D interaction. The discussion relies on our experiences developing the QuickSet system and integrating it with the Naval Research Laboratory's (NRL) Dragon 2 VR System on a Responsive Workbench and in a fully immersive display similar to a Cave Automated Virtual Environment (CAVE) called a "Grotto."

QuickSet

QuickSet is a wireless, handheld, agent-based, collaborative, multimodal system for interacting with distributed applications. The QuickSet user holds or wears a small computer displaying an interactive map (see Figure 1). The user can speak and draw to create entities on the map and to control a number of back-end systems, including 3D visualization systems. The system analyzes continuous speech and gesture in real time, producing the best joint semantic interpretation for multimodal commands.

QuickSet consists of a collection of "agents" including speech recognition, gesture recognition, natural language understanding, multimodal integration, a map-based user interface, and a database, running standalone on the tablet PC or distributed over a network. The multimodal interface runs on machines as small as Windows CE devices, as well as on wearable, handheld, table-sized, and wall-sized displays. The system components are integrated via the Open Agent Architecture (SRI International), which offers facilitated communication, plug-and-play connection, dynamic discovery of agents, asynchronous operation, and "wrapper" libraries in C++, Prolog, Java, and other languages. A diagram of the QuickSet architecture appears in Figure 2. Because of QuickSet's agent architecture, we can easily incorporate other sources of digital "ink" besides an electronic stylus, plus other devices for signaling the onset of speech.

We've integrated QuickSet with three visualization systems: CommandVu from the US Navy's Space and

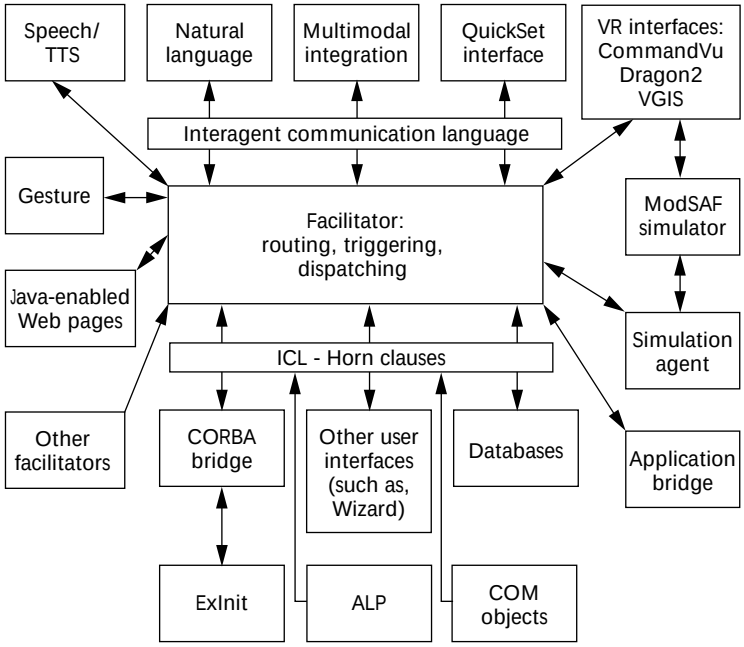
0272-1716/99/$10.00 © 1999 IEEE

Naval Warfare Systems Command (SPAWAR), Virtual Geographic Information System (VGIS) from the Army Research Laboratory and Georgia Institute of Technology, and more recently with NRL's Dragon 2. A QuickSet user can create entities in a Modular Semi-Automated Forces (ModSAF) distributed simulation, assign missions, and watch the simulation unfold on the handheld PC and the 3D visualization platform. The user can also issue spoken or multimodal commands using the QuickSet 2D interface, which the visualization systems then execute.

Sample commands include

1. "Turn on heads up display."
2. "Take me to objective alpha."
3. "Fly me to this platoon" accompanied by a gesture on the QuickSet map.
4. "Fly me along this route at fifty meters" accompanied by drawing a route on the QuickSet map.

Here, users (1) control the visualization, (2 and 3) navigate to out-of-view locations and entities, and (4) move along prescribed paths. To explain the process, we turn to QuickSet's multimodal integration architecture.



2 QuickSet's facilitated architecture.

Multimodal integration architecture

We can summarize the multimodal integration process for QuickSet as follows. The system employs continuous speech and continuous gesture recognizers running in parallel. It supports a wide range of continuous gestural input, including points, lines, areas, various types of arrows, and military symbols.

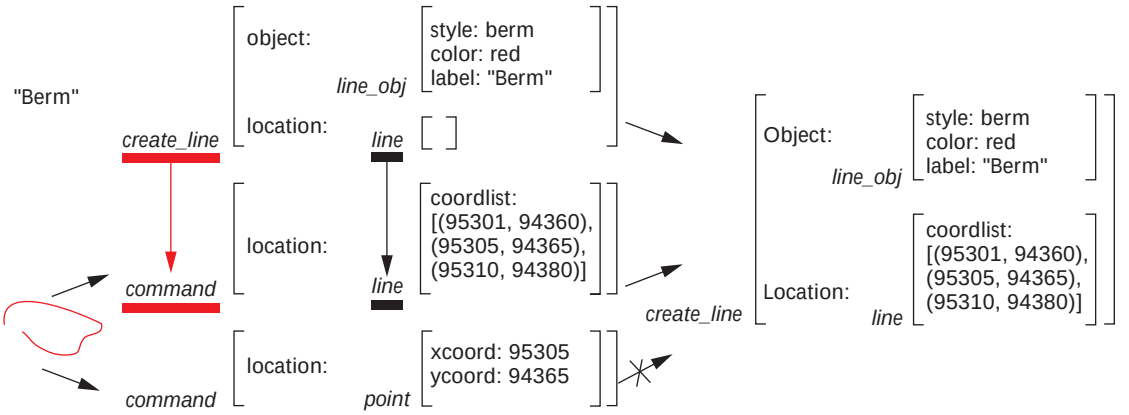
- Typed "feature structures" provide a clearly defined and well-understood common meaning representation for the modes.
- Multimodal integration is accomplished through unification.
- The integration is sensitive to the temporal characteristics of the input in each mode.
- Unlike systems with a control strategy dominated by the spoken language parsing process, QuickSet's multimodal processing responds to either mode. Thus, the system supports unimodal speech or gesture, as well as multimodal input in which no deictic term (such as "here" or "this") occurs.
- A statistical unification-based integration method lets spoken language and gesture compensate for recognition errors in either unimodal technology.
- Requests for confirmation of the system's interpretation occur after multimodal integration. This lets mutual disambiguation correct errors and obviates the need for users to do so.
- The agent architecture offers a flexible asynchronous framework within which to build multimodal systems.

Example of 2D multimodal interaction

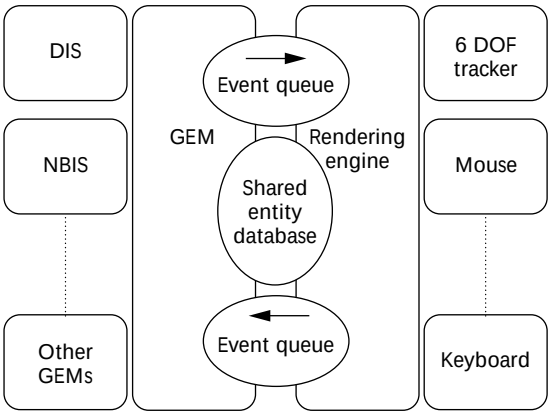
Holding QuickSet, the user views a map from a ModSAF simulation. Using spoken language coupled with pen gestures, the user adds entities into the ModSAF simulation and gives them behavior. For example, to create a platoon in QuickSet, the user would hold the pen at the desired location and say, "M1A1 platoon." The user then adds a berm to the simulation by drawing a line at the desired location while uttering "berm." Adding a fortified line multimodally only requires that the user draw a simple line and speak its label, or draw its military symbology. The user can draw and label a "no go area" of amorphous shape verbally. Finally, the user can assign a task to the new platoon by saying "M1A1 platoon follow this route" while drawing a line.

As an example of multimodal integration, consider processing of the utterance "berm" while drawing a line. Processing of the speech results in a set of feature structures, one of which hypothesizes that the user is creating a line whose location is unspecified, whose style is berm, and is labeled "Berm." One gesture hypothesis asserts that the user is performing a line command with the drawn set of coordinates. A second hypothesis asserts that the user is performing a command with a point at the centroid of the drawing. The typed feature structure unification process unifies the two `line` feature structures, provided that `create_line` is a subtype of command. (See Figure 3, next page.) Overall, the integration process examines the cross product of speech and gesture interpretations, subject to temporal constraints, and ranks the successfully unified ones according to their joint probability. We are currently developing a more sophisticated statistical unification process.

In summary, unification proves a good candidate for the information fusion operation, since it allows combining complementary and redundant information from each mode, but disallows conflicting information. This process can support mutual disambiguation of both



3 Unifying speech with two gestural interpretations for "berm."



4 The Dragon 2 software architecture.

input signals, in that it is possible to derive the highest ranking multimodal interpretation from spoken and/or gestural interpretations that are not the highest in their category. In such cases, one mode has compensated for an erroneous recognition in the other.

Evaluation

We designed QuickSet based partly on proactive empirical studies using a "Wizard of Oz" paradigm. Recent studies with QuickSet have shown the value of multimodal systems offering mutual disambiguation of recognition inputs over unimodal speech or gesture processing. Our studies have also demonstrated substantial efficiency advantages of multimodal interaction over GUIs for map-based tasks. For example, a recent case study of a US Marine Corps major compared the time taken to create and position military entities on a map using a commercial GUI and using Quickset, with the same back-end system. Results showed a three- to ninefold speed improvement when using multimodal interaction versus the GUI.

Multimodal interaction for 3D systems

We believe that multimodal interactions will form an extremely powerful paradigm for interaction in 3D environments. By providing extra degrees of freedom, 3D environments offer a much richer set of interactions. At the same time, they introduce many complexities into the design of the user interface. We hypothesize that users will function more efficiently with multimodal interaction than with standard direct-manipulation styles of 3D interaction. To investigate this hypothesis, NRL and OGI have begun to explore extending QuickSet's multimodal interactions into 3D. Specifically, we integrated Dragon 2 and QuickSet.

Dragon 2

NRL developed the Dragon 2 system as a research platform to study the design and use of VR systems for battlefield planning and control applications. Figure 4 illustrates the system's architecture. It consists of two tightly coupled subsystems: the Generic Entity Manager (GEM) and the Rendering Engine (RE). The GEM collects data from external data sources and expresses them in a common, standard representation. The RE implements the user interface. It draws the virtual environment, processes user input, and creates a set of requests and events directed back to the GEM. The two subsystems interact through a pair of unidirectional event queues (specifying the state changes that have occurred) and a shared entity database (specifying new entity data). We've implemented external interfaces to a range of systems, including ModSAF and other systems communicating via the Distributed Interactive Systems (DIS) protocol.

Integrating an interface into the Dragon 2 system proved relatively straightforward. For Dragon 2, the Open Agent Architecture acts like another data source. For the Open Agent Architecture, Dragon 2 is merely another user interface agent.

Direct multimodal interaction with 3D environments

The 3D interaction paradigm was designed to work on a Responsive Workbench or in a Grotto.

NRL has developed voice and "touchglove"-based interaction methods, but for fielded systems NRL uses a 3D joystick. The custom 6DOF "flight stick" consists of a commercial joystick modified to incorporate a Polhemus tracker. Various modes permit using the flight stick for navigation, viewpoint control, selection, and drawing of "digital ink." The flight stick appears as a beam of light emanating from the approximate position of the user's hand in the virtual world.

Pressing a button on the flight stick while in draw mode lets the user draw digital ink by casting a ray on the virtual 3D terrain. Where the ray touches the terrain's surface, the system deposits a trail of "ink balls." The user can manipulate the size and the minimum distance between successive ink balls through configuration file settings. The ink information then goes as a sequence of latitude/longitude coordinates to the facilitator for routing to the gesture recognition agent.

The same button event also starts the speech recognition process. Speech and gesture are recognized in parallel, parsed, then fused via the QuickSet multimodal integration agent. The resulting entity creation messages return through the agent architecture, where Dragon 2 processes them and renders them onto the terrain. Users thus can create and position entities, such as platoons of vehicles, or draw control measures, by speaking while gesturing in 3-space. When linked with other user interfaces through the same facilitator, Dragon 2 can show the ink laid by remote users and any objects created (see Figure 5). Future versions of the integrated system will provide modeless gestural and selection capabilities, as found in QuickSet.

The next major step involves integrating the semantics of spoken language expressions (such as "hill" in "how high is this hill <3D gesture>") with knowledge from the scene graph. Here, a topographical recognizer needs to determine that the part of that graph intersected by the 3D gesture represents a hill. How best to integrate such recognizers into this process represents an active area of our research. We'll also assess the extent to which we can optimize mutual disambiguation of speech and 3D gestures within our statistically based multimodal architecture to benefit 3D multimodal interaction.

A practical goal of our research is to build a system in which users can both speak and gesture into 3D projected images with a tracked laser pointer. This gestural method allows interaction with "Power Walls"—very large screen display surfaces. QuickSet already runs with a laser pointer serving as a 2D drawing device (see Figure 6). We plan to evaluate the usability of laser-based interaction in comparison to other 3D gesture devices. ■

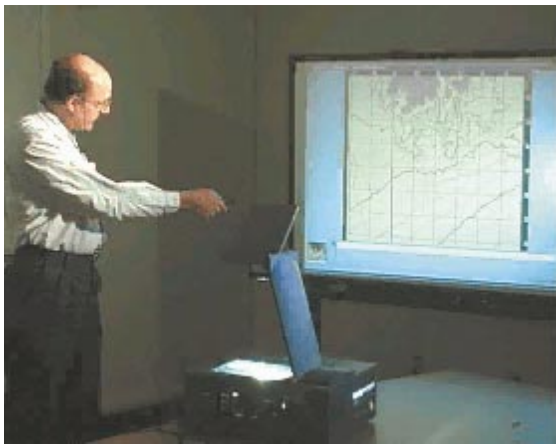
Acknowledgments

OGI's work is supported in part by the Human-Computer Interaction Program of Darpa under contract number DABT63-95-C-007, and Darpa's Command Post of the Future Program, contract number N66001-99-D-8503; also, in part, by ONR grants: N00014-95-1-1164, N00014-99-1-0377, and N00014-99-1-0380. NRL's research is supported by ONR. Portions of OGI's research have been conducted in collaboration with the SPAWAR Systems Center.

Contact Cohen by e-mail at pcohen@cse.ogi.edu.

Contact department editors Rosenblum and Macedonia by e-mail at rosenblum@ait.nrl.navy.mil and macedonia@computer.org.

5 Applying digital ink on a workbench display.



6 Operating QuickSet with a camera-tracked laser pointer.

Further Reading

P.R. Cohen et al., "An Open Agent Architecture," *Working Notes of the AAAI Spring Symp. Series on Software Agents*, held 21-22 March 1994 at Stanford Univ., Calif., American Association for Artificial Intelligence, Menlo Park, Calif., pp. 1-8. Reprinted in *Readings in Agents*, M. Huhns and M. Singh, eds., Morgan Kaufmann Publishers, San Francisco, 1998.

P.R. Cohen et al., "QuickSet: Multimodal Interaction for Distributed Applications," *Proc. ACM Multimedia*, ACM Press, New York, 1997, pp. 31-39.

S. Julier et al., "The Software Architecture of a Real-Time Battlefield Visualization Virtual Environment," *Proc. IEEE VR 99*, IEEE Computer Society Press, Los Alamitos, Calif., March 1999, pp. 29-36.

S.L. Oviatt, "Multimodal Interactive Maps: Designing for Human Performance," *Human-Computer Interactions*, Vol. 12, 1997, pp. 93-129.

L.J. Rosenblum et al., "Situational Awareness Using the VR Responsive Workbench," *IEEE CG&A*, Vol. 16, No. 4, July/August 1997, pp. 12-13.