

# BattleView: A Multimodal HCI Research Application

Gregory A. Berry, Vladimir Pavlović<sup>1</sup> and Thomas S. Huang  
Image Formation and Processing Group, Beckman Institute  
ECE Department and Coordinated Science Laboratory  
University of Illinois at Urbana-Champaign  
405 N. Mathews Avenue, Urbana, IL 61801  
{*berry,vladimir,huang*}@*ifp.uiuc.edu*

<sup>1</sup>**Corresponding Author:** tel. (217) 244-1089, fax (217) 244-8371

## Abstract

*To demonstrate some of our research topics in Human Computer Interaction (HCI), we employ two modes of natural human-computer interaction to control a virtual environment. By using speech and gesture recognition, we outline the control of a virtual environment research testbed (BattleView) without the need for traditional virtual reality interfaces such a wand, mouse, or keyboard. The use of features from both speech and gesture creates a unique interface where different modalities complement each other in a more “human” communication style.*

## 1. Introduction

Since the advent of the computer, the user has been forced to conform to the interface dictated by the machine. For the ancient Chinese the interface was the beads of the abacus. In the 1960s the keyboard of the punch card machine and teletype constrained the user to batch jobs and blinding print speeds. The arrival of interactive text terminals in the 1970s was a noticeable leap, but soon even typing was seen as a burden, and a more efficient interface was developed. Graphical operating systems of the 1980s, inspired by the “look-and-feel” of a desktop, introduced the mouse—a simple pointing device for the user. In the 1990s, with increases in computational power, basic speech recognition and pen-based computing has become a reality.

All the aforementioned interfaces, with possible exception of the speech recognizer, have been efficient for a trained user. However, they are typically inefficient as a human-centric form of communication. With recent advances in Human Computer Intelligent Interaction it has become more feasible to create interfaces that resemble forms of human communication. Although it is still impossible to create a ubiquitous interface that can handle all forms of human communication, it is possible to create a small multimodal subset. Most modern interfaces rely exclusively

on a single mode of interaction such as speech or mouse, but rarely attempt to use multiple modes. To make the design of multimodal interface manageable, rather than define a generic interface, we constrain the interface by the task. Therefore, we limit the modes and commands, auditory and visual, to the specific domain.

## 2. Motivation

For our application we chose to implement an interface for a virtual reality environment called BattleView. BattleView is a creation of the National Center for Supercomputing Applications (NCSA) for studying graphic display and user interaction strategies in support of planning and decision-making tasks in a virtual battlefield [2]. The user is immersed in a display of terrain populated with battlefield objects, such as tanks and planes, which he can manipulate. By definition, a virtual reality environment is a computer-generated representation of three-dimensional space that enables a user to experience an immersive, interactive simulation of an imaginary or realistic environment. To complete the immersive experience the human-computer interface must be simple and natural. If the environment is hard to use or a distraction to the user, the immersive quality of the environment dramatically erodes.

Recently systems have emerged that integrate multiple interface modalities in desktop and hand-held computer environments [6, 13, 3]. A leap away from small-scale interactive virtual environment has been made at M.I.T. Media Lab through their Perceptual Spaces [14] and Unencumbered Virtual Environments [11]. Another perfect yet rarely explored testbed for multimodal interface is in large-scale immersive displays or virtual environments such as the CAVE [4] and Immersadesk technologies [1]. Our research efforts have been aimed at introducing natural multimodal interface to these established virtual domains.

### 3. BattleView

We chose to implement BattleView on an Immersadesk2 [1] system. Interaction with the environment of an Immersadesk is typically accomplished via magnetic tracked position sensors and a pointing device called a *wand*. The *wand* is essentially a 3D mouse with a receiving antenna attached so that the computer constantly receives information about the wand's position and orientation. The current wand has three buttons and a pressure-sensitive joystick. The joystick is used primarily for navigation in combination with the position and orientation information. The buttons are used to set modes and select options. In its present form the wand distracts from the naturally immersive display, because the user remains tethered to the computer. Hence by freeing the user from the wires, and using hand gesture and speech recognition the human-computer interface becomes more human-centric.

#### 3.1. Hand Gesture

Hand gestures by definition are the motions of the hands to express ideas or to generally communicate. People commonly use hand gesture to emphasize important points during discourse, or to express ideas or spatial relationships that are hard to verbalize. For example, when describing the size of a catch, a braggart is likely to gesture a distance with his hands rather than say that the fish was 27.3 centimeters in length. Hence, hand gestures tend to provide a natural communication mode that can be exploited.

Machine recognition of the human hand gestures has attracted increased interest in the last several years. Numerous approaches ranging from the ones that use specialized gloves to free-hand visual interpretation have been attempted (see [10] for a recent review of hand gesture analysis and interpretation techniques.) Original work of hand gesture recognition and visual computing environments in the Image Formation and Processing Group at the University of Illinois was done by J. Kuch [5] and V. Pavlovic [9]. Both of these authors' work dealt with the vision based gesture recognition. Also, their work strives to make a more natural Human Computer Interface. The vision based approach allows the user to be free of direct contact with the computer. In other words, there are no wire tethers, keys, or glove to distract the user. Thus, BattleView's gestural interface employs a vision based input.

BattleView uses a method of real-time color- and motion-based hand tracking (see [8]). Fusion of the color and motion cues provides a simple yet robust way of detecting and tracking the user's hand(s). The simple algorithms are fast enough to run in real-time on a standard Silicon Graphics O2. To capture video data a single camera is mounted on the top center of the Immersadesk screen (see Figure 1). The field of view of the camera is pointed down



Figure 1. BattleView virtual environment on an Immersadesk 2.

vertically such that the users hand is easily detected. Alternatively, a set of two cameras mounted on the two sides of the Immersadesk screen is used in the stereoscopic version of the hand tracker.

#### 3.2. Speech Recognition

For the auditory module, rather than develop our own word level recognizer as in [12], we use a commercially available *ViaVoice* continuous speech recognition system from IBM. The reasons for using the commercial system rather than our own custom system are twofold: the BattleView environment supports speech recognition through this package, and the commercial system allows for rapid prototyping for word level recognition. A set of command words and a simple grammar were incorporated into the auditory module. The command words are depicted in Table 1. Audio data is captured using a head mounted wireless

Commands	Start, Stop, Go(ing) / Move(ing), Backward, Forward, Left, Right, Slower, Faster, Select (this), Show information about this object Weapon hit
----------	---------------------------------------------------------------------------------------------------------------------------------------------------------------

Table 1. Speech commands for BattleView.

microphone (see Figure 1).

#### 3.3 Integration of Modes

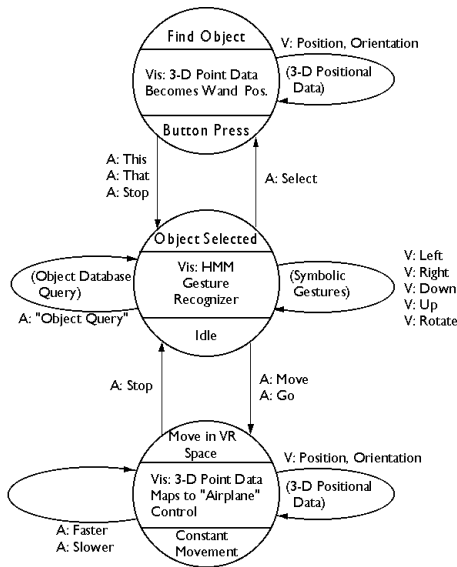
Integration of gestures and speech occurs in a multimodal integration module. In this module, output data from the audio and visual modes is combined in both *complementary* and *reinforcing* fashions to create an input into BattleView. Multimodal data fusion architectures in the HCI

domain usually combine frame-based fusion supplemented by multi-agent systems [6, 13, 3]. We borrowed some basic tools from the above approaches and encapsulated them into a finite-state grammar appropriate for the system.

The integration module in BattleView emulates and significantly enhances the wand functionality of the original Immersadesk setup. Namely, it facilitates:

- Terrain navigation,
- Object selection, and
- Object manipulation.

The module’s functionality is defined by a finite state machine depicted in Figure 2. Each state has three sections which summarize the state’s identity and actions. The top one third section of the state’s representation denotes the name of the state. The center section of the state defines the state of visual module’s gesture recognizer. The visual module can interpret two different forms of gestures, and the center section tracks which gesture form to use. Lastly, the lower third section of the state documents the wand function emulated by the state. In Figure 2 the arcs with



**Figure 2. Integration Module State Machine. The “A: <command>” represents verbal commands. The “V: <command>” represents gesture commands.**

“V: <command>” correspond to visual module commands or data, and the arcs with “A: <command>” correspond to audio commands. The commands can change a state or modify data.

The state machine starts in the idle **Object Manipulation** mode (“Object Selected” state), with no identified ob-

jects. For the user to start movement in the virtual environment, he or she utters phrases such as “start moving forward” or “go backward.” When the verbal command is recognized and forwarded to the integration module, the visual module’s gesture recognizer changes from a feature classifier to a simple hand position tracker. Thus, after the state change, the visual module would provide position and orientation information to BattleView (**Terrain Navigation** mode). In the standard BattleView VR environment, the position and orientation data would be supplied by the wand, but in our application the hand tracking system provides the data. Using the position of the hand with reference to the head, the gesture now defines an airplane-like “stick” control. To change the speed and direction in standard BattleView the user adjusts a joystick, but in this interface, the user utters “faster” or “slower” (or “go faster” and “go slower”). In Figure 2 the action is noted as the arc “A: Faster” on the “Move in VR Space” state. Again, the verbal command modifies the state of the application. The role of speech and hand motion in this case is *complementary*.

To return to the **Object Manipulation** the user would utter “stop” or “stop moving.” In the Object Manipulation state, the visual module’s gesture recognizer facilitates symbolic gesture and speech recognition. The visual module can respond to simple hand gestures to orient the user with respect to the object. Hence, if the user makes a “left” command gesture, the view point moves to the left. The user can *reinforce* any gestural command with a corresponding spoken command. Thus, gestural “move left” can be accompanied by a spoken “move left”. The fusion of the two modalities for reinforcement is carried out by selecting the *jointly most likely speech-gesture pair*, constrained by the grammar and the overlapping of action time intervals. In addition, purely verbal queries such as “show information about this object” and actions such as “weapon hit” can be invoked in this state. Object Manipulation mode is disengaged by uttering “release” or moving away from the object’s sphere of interaction.

To select a new reference object, the user utters “select.” The system then transitions into the **Object Selection** mode where deictics and speech play *complementary* roles. Here a 3D picking ray is “attached” to the user’s hand and enables him or her to point at an object on the terrain. Object selection is finalized by uttering “this”<sup>1</sup>. The described action takes the module back into the **Object Manipulation** mode where further manipulation or querying is possible.

#### 4. System Evaluation and Conclusions

Introduction of bimodal (speech and gesture) interface into BattleView has been considered a success. Judged by

<sup>1</sup>Visual feedback is provided in the form of a highlighted bounding box indicating that an object is being pointed at.

numerous untrained and expert users the system is very easy to use, quite natural, and fairly reliable. When used in demonstrations to the public, untrained users almost instantaneously understood how to use the interface. Two aspects proved to be crucial for the usefulness of the bimodal interface: real-time performance, and presence of visual feedback. Both of those aspects alleviated the inherent deficiencies of the hand tracking and the speech recognition systems and made them transparent to the user.

A major benefit of the current system is that its components (such as gesture and speech recognition units and multimodal unit) are *easily modifiable*. This allows us to use the system as the testbed for novel multimodal fusion architectures with minimal system building effort.

Currently, the gesture recognizer only responds to pointing gestures, and it doesn't recognize gestural commands on-line. However, the gestural command recognition has been demonstrated in a separate application to yield sufficient accuracy, and plans are underway to incorporate it into the system. Our present research efforts are focused on introduction of natural language / probabilistic modeling processing elements (verified in off-line architectures, such as [7]) in the on-line integration module.

## References

- [1] Virtual Reality Systems at Electronic Visualization Lab (EVL), University of Illinois at Chicago. <http://www.evl.uic.edu/EVL/VR/systems.html>.
- [2] P. Baker. Battleview. <http://www.ncsa.uiuc.edu/Vis/Projects/BattleView>.
- [3] P. Cohen, M. Johnston, D. McGee, S. Oviatt, and J. Pittman. QuickSet: Multimodal interaction for simulation set-up and control. In *Proceedings of the 5th Applied Natural Language Processing Meeting*, Washington, DC, 1997. Association of Computational Linguistics.
- [4] C. Cruz-Neira, D. Sandin, T. DeFanti, R. Kenyon, and J. Hart. The cave: Audio visual experience automatic virtual environment. *Communications of the ACM*, pages 65–72, June 1992.
- [5] J. Kuch. Vision-based hand modeling and gesture recognition for human computer interaction. Master's thesis, University of Illinois at Urbana-Champaign, Urbana, IL, Sep. 1994.
- [6] L. Nigay and J. Coutaz. A design space for multimodal system: a concurrent processing and data fusion. In *INTERCHI'93*, pages 172–178, April 1993.
- [7] V. Pavlovic. Multimodal tracking and classification of audio-visual features. In *Proceedings of the IEEE International Conference on Image Processing*, Chicago, IL, October 1998.
- [8] V. Pavlovic, G. Berry, and T. Huang. Fusion of audio/visual information for human-computer interaction. In *Proceedings of the Workshop on Perceptual User Interfaces*, pages 69–71. IEEE, Oct. 1997.
- [9] V. Pavlovic, R. Sharma, and T. Huang. Gestural interface to a visual computing environment for molecular biologists. In *Proceedings of the IEEE Conference on Automatic Face and Gesture Recognition*, pages 30–35, Killington, VT, October 1996. IEEE.
- [10] V. Pavlovic, R. Sharma, and T. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 19(7), 1997.
- [11] K. Russel, T. Starner, and A. Pentland. Unencumbered virtual environments. *International joint conference on artificial intelligence workshop on entertainment and AI/Alife*, 1995.
- [12] R. Sharma, T. Huang, V. Pavlovic, K. Schulten, A. Dalke, J. Phillips, M. Zeller, W. Humphrey, Y. Zhao, Z. Lo, and S. Chu. Speech/gesture interface to a visual computing environment for molecular biologists. In *Proc. International Conference on Pattern Recognition*, Vienna, Austria, Aug. 25-30 1996.
- [13] A. Waibel, M. T. Vo, P. Duchnowski, and S. Manke. Multimodal interfaces. *Artificial Intelligence Review*, 10(3-4):299–319, August 1995.
- [14] C. Wren et al. Perceptive spaces for performance and entertainment: Untethered interaction using computer vision and audition. *Applied Artificial Intelligence*, 11(4):267–284, June 1997.