Interacting Naturally in Virtual Environments

David E. Diller, Robert F. Stark, Brian Krisler,
David McDonald, Richard Shapiro, Kerry Moffitt

Raytheon BBN Technologies

Cambridge, MA

ddiller@bbn.com, rstark@bbn.com, bkrisler@bbn.com,
dmcdonald@bbn.com, rshapiro@bbn.com, kmoffitt@bbn.com

ABSTRACT

Current methods for controlling one's avatar in a virtual environment interacting with intelligent virtual agents (IVAs) are unnatural, typically requiring a complex set of keyboard commands for controlling your avatar, and dialog menus for interacting with IVAs. Recent advances in markerless body and motion tracking, speech and gesture recognition technologies, coupled with intelligent agent/behavior modeling and speech synthesis technologies, now make it possible to naturally control one's avatar through the movement of one's body and to interact with IVAs through speech and gesture. These capabilities are now just beginning to emerge in the arena of computer gaming, and offer great promise for military training. In this paper we describe our recent work integrating motion capture, gesture recognition, speech recognition, natural language understanding, and intelligent agent/behavior modeling technologies to produce more natural mechanisms for avatar control as well as IVAs that are able to understand relatively unconstrained speech and recognize human movement and gesture. We illustrate these capabilities within the domain of roadside security checkpoint training, where trainees are able to gesture (e.g., wave forward, stop, point to a location) and speak to IVAs (drivers and passengers) in the scene.

ABOUT THE AUTHORS

David E. Diller is a Senior Scientist at Raytheon BBN Technologies in Cambridge, MA. He holds a B.S. in Computer Science and Psychology from Taylor University, an M.S. in Computer Science and a joint Ph.D. in Cognitive Science and Cognitive Psychology from Indiana University. His research interests include cognitive modeling, mixed-initiative agent-based systems, and simulation-based training applications.

Robert F. Stark is a Staff Scientist at Raytheon BBN Technologies in Cambridge, MA, where he develops training simulations and other intelligent and usable software systems. He holds an MS in Information Sciences and Technology from Penn State University with a thesis on the human factors benefits of multimodal input for military training simulations.

Brian Krisler is a Scientist at Raytheon BBN Technologies in Cambridge, MA. He holds an M.S. in Computer Science from, and is currently a PhD candidate at Brandeis University. His research interests are cognitive skill acquisition and human-computer interaction.

David McDonald is a Senior Scientist at Raytheon BBN Technologies in Cambridge, MA. He has more than thirty years experience in artificial intelligence and computational linguistics research and development. He is the author of over sixty refereed publications with contributions in the fields of natural language generation, robust semantic parsing, knowledge representation, and synthetic agents.

Richard Shapiro is a Senior Scientist at Raytheon BBN Technologies in Cambridge, MA. He has worked as a software engineer for over thirty years, across a wide range of platforms, languages, and programming paradigms. He has a particular interest in the aesthetics of component integration.

Kerry Moffitt is a Scientist at Raytheon BBN Technologies in Cambridge, MA. He has developed real-time games – for entertainment and for training – for over fifteen years. He has also programmed communications software and graphical authoring tools, and studies pedagogy and aesthetics.

Interacting Naturally in Virtual Environments

David E. Diller, Robert F. Stark, Brian Krisler,
David McDonald, Richard Shapiro, Kerry Moffitt

Raytheon BBN Technologies

Cambridge, MA

ddiller@bbn.com, rstark@bbn.com, bkrisler@bbn.com,
dmcdonald@bbn.com, rshapiro@bbn.com, kmoffitt@bbn.com

INTRODUCTION

Virtual environments (VEs) and the avatars that populate them are becoming increasingly rich and complex, both visually, and in their capacity for action and control. Consequently, a complex and unnatural set of commands must be learned in order to control an avatar in VEs. Similarly, communications with intelligent virtual agents (IVAs) are currently often quite limited and unnatural.

The available command sets for puppeteering an avatar are outstripping our ability to make effective use of them using standard controllers such as a keyboard & mouse or game controller. In tactically oriented VEs, avatars can typically walk, run, crouch, crawl, lean, and look around – and a different keyboard shortcut must be memorized for each action. Additional commands are needed to trigger gesture animations and use equipment (e.g., weapons, binoculars, vehicles, etc.). The default control set for one tactically oriented VE, Virtual Battlespace 2 (VBS2), now has over 30 possible avatar movement/action controls. In more socially oriented virtual environments, the desire for fine-grained control of an avatar is even greater because participants wish to communicate emotional state non-verbally to other inhabitants. For example, in Second Life, avatars can not only walk, but also fly, and perform a rich set of gestural animations (130+), including facial gestures (frown, wink, etc.). A shift is required from standard keyboard and mouse input control devices to other more intuitive and natural interface mechanisms.

Although the need for and desirability of IVAs is widely accepted, their use has been hampered by limitations in their ability to communicate effectively and naturally with humans, and to exhibit realistic and robust behaviors. IVAs often are implemented with highly scripted context-specific responses, and users interact with them through dialog trees, selecting from a small set of preprogrammed choices. To circumvent these issues, the US Military often makes extensive use of human role players. However, training using human role players is expensive to conduct, time consuming to set up, and requires skilled personnel that can be difficult to recruit and retain. Once IVAs are able to communicate more effectively and naturally with humans, and to exhibit more robust, adaptive behaviors, they will hold great promise as a solution to these problems.

New technologies, coupled with reductions in costs of existing technologies, hold great promise for changing the way we control avatars and interact with IVAs in virtual environments. By leveraging technologies such as inexpensive gyroscopes, accelerometers, and pressure sensors for head, body, and motion tracking (often bundled into low cost game controllers (e.g., Wii remote), markerless body and motion tracking using increasingly low-cost 3D time-of-flight (TOF) cameras, gesture recognition, natural body movements and gestures can be translated into virtual events for communication and control. Combining these actions with other natural interfaces such as speech will not only improve avatar control and communication, but also make the system easier to learn and use effectively, and provide a more natural immersive experience. As a result, we expect this would improve the users' performance and training, and in turn, improve transfer of learned knowledge and skills to the real world.

Current advances in technologies for avatar interaction and control are largely being driven by the competitive video-game industry. The success of Nintendo's Wii has led both Sony and Microsoft to announce upcoming new game-controllers with gesture recognition capabilities. Microsoft has announced Kinect™[1] (formerly Project Natal) for the Xbox 360, which utilizes a 3D TOF camera for tracking body motion and also includes speech recognition capabilities. Sony announced PlayStation® Move[2], which combines hand-held controllers with a camera to track player motion.

[1] http://www.xbox.com/kinect

[2] http://us.playstation.com/ps3/playstation-move/

Our research explores the degree to which interfaces for virtual environments can be made natural and intuitive while leaving the user relatively unencumbered, and requiring little hardware. This work builds on a history of research in multimodal interfaces, much of which started with a system called "Put-That-There" (Bolt, 1980) that allowed a user to point to a screen and say commands such as "put that there", which would move the shape pointed at to the second location indicated by pointing. More recently, work has integrated this type of interaction with virtual agents (e.g., Maes et al., 1997; Demirdijian, Ko, & Darrell, 2005), although not focused on the particular needs of training systems. With the needs of military training in mind, we hope to provide the users with the ability to use their own social awareness and skills, such as gesturing and speech, coupled with use of the their body awareness and skills to facilitate navigation within virtual environments (Jacob et al., 2008).

To demonstrate the capabilities of the IVAs and avatar control, we developed a prototype checkpoint security trainer that allows a trainee to participate by controlling an avatar situated in a 3D virtual environment. This prototype includes a light-weight solution for IVAs that are capable of standing in for human role players. These IVAs can interact naturally with humans by understanding and generating speech and gesture, and demonstrating believable behaviors based on the context of the interaction. To achieve this level of natural interaction, we integrated real-time speech recognizers (Boisen, et al., 1989; Stallard, et al., 2007), statistical grammars, natural language understanding, motion tracking and gesture recognition. Coupled with rules-based intelligent agent/behavior modeling technologies and speech synthesis our IVAs can respond and react to a large corpus of domain specific questions and directives. To combine these inputs, the prototype also uses state-of-the-art multimodal fusion to allow the speech and gestures to complement each other (e.g., Chai et al., 2002).

In the role of a soldier manning a security checkpoint, the user can perform typical traffic control requests (i.e. *drive forward*), tactical, domain specific IVA questioning (*Who is with you?, Open the trunk.*) and scene-specific, multimodal directives (*Pointing to a location*. "*Please go stand over there"*) through the use of speech and gesture.

Figure 1: A user interacts with the system by standing in front of a large screen displaying the 3D virtual environment. The user's view is the first person perspective of their avatar.

Navigation within the virtual environment occurs by positional shifts performed by the user: stepping forward to move forward or backward to move back, leaning left to lean left, etc. The user's movements are tracked by a single camera mounted above the screen (see Figure 1), interpreted, and then transmitted as navigational directives or communication events to the virtual environment, or IVAs, respectively. All non-gesture based interaction in the environment occurs via a microphone and a Wii remote.

SYSTEM ARCHITECTURE

The system receives inputs from a number of different devices, including a single 3D time-of-flight camera; a microphone for capturing voice communication, and a Wii remote for changing point of view, manipulating the game engine action menu, firing a weapon, and for triggering voice audio capture (i.e., voice push-to-talk). The architecture (see Figure 2) integrates a number of subsystems, including a voice-over-IP audio communications system, body motion tracking, and a virtual environment, in addition to the intelligent virtual agent components. While all subsystems have been designed to run on separate hardware platforms, our standard method of implementation is to run all components on a single computer.

The virtual environment utilized was VBS2. VBS2 is a multi-player, game-based training system widely used by the US Military.

Intelligent virtual agents consist of a set of interacting modules, including:

- *Gesture Recognition:* Recognizes gestures from a stream of data describing the position of various parts of the body (e.g., hands, head, shoulders).
- *Speech Recognition:* Two speech recognizers, a grammar-based recognizer, and a statistically-based recognizer.
- *Language Parsing:* A semantic text analyzer which converts a speech recognition transcript into the semantic representation utilized by the behavior engine.
- *Speech Generation:* A commercial text-to-speech engine.
- *Behavioral Rule Engine & Fact Base:* Behavioral modeling architecture incorporating the Java-based Drools Production Rule System (Bali, 2009).
- *Action Scheduling & Generation:* A subcomponent of the behavioral rule engine which manages goals and actions to accomplish those goals.
- *World State Monitoring:* Selectively monitors relevant world-state from the game engine and updates the fact base used by the behavioral model.

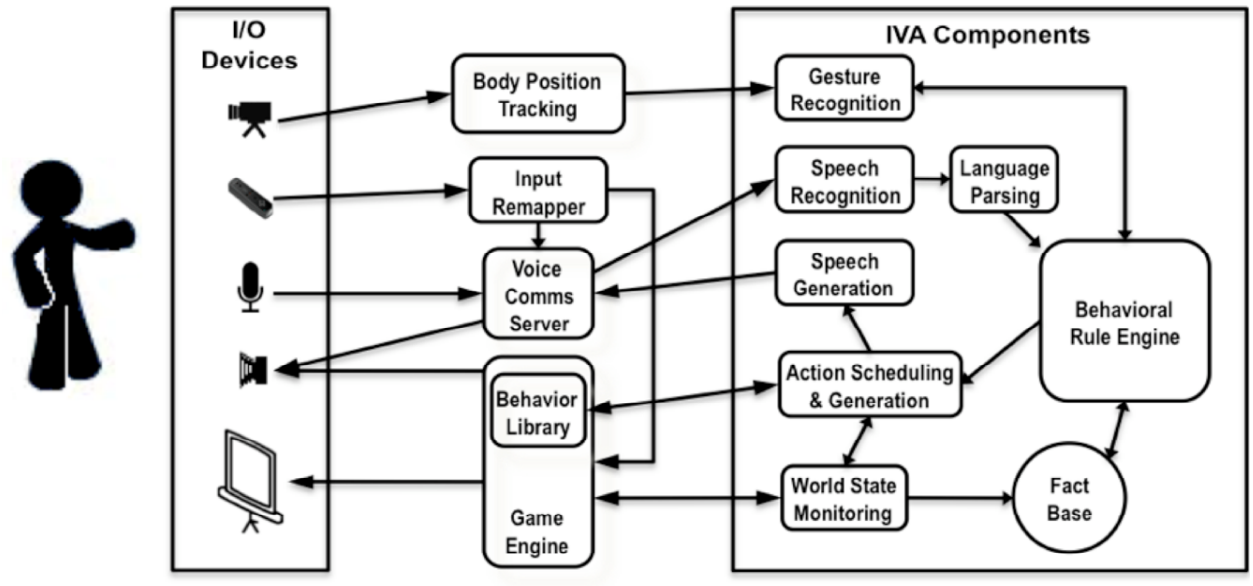

Figure 2: System Architecture

AVATAR CONTROL

Using a 3D TOF camera, coupled with software for tracking user body position, the user's body acts as the controller. This is a central component of the system.

When the user recognizes that a person has stepped into the scene, the user is instructed to stand still briefly while the user's body position and posture are recorded. This information is then used as a baseline against which to compare future movements by the user.

One of the control mechanisms implemented is the usage of moving in various directions to control avatar movement. When the user steps away from the neutral center position, it is recognized as a move in that direction, and so the system moves the avatar in that same direction. Moving farther in a direction changes the avatar's speed from walking to running, and moving back to the original center position halts the avatar's movement. The avatar's directional movements mirror the user's movements. Step left to move the avatar left, back to move the avatar back, diagonally to move diagonally, etc.

The other control mechanism is leaning. Because the system knows what the at-rest posture is, when the user changes that posture by leaning to the side, the system has the avatar lean accordingly.

At the time of development, VBS2 did not provide an application programming interface (API) for skeletal control of an avatar. To address this issue, we created a mapping of navigational queues to the appropriate VBS2 keyboard events. These keyboard events were sent via an event generator to the VBS2 system, simulating the key presses. One of the benefits of keyboard event generation is the allowance for easy deployment to other virtual environments simply by changing the mapping between user actions and the keyboard events that trigger the same avatar actions. However, virtual environments that possess APIs supporting full avatar joint control (e.g., Second Life) provide a richer and more robust method for avatar control, eliminating the need for keyboard event generation.

INTERACTING WITH IVAs

Interacting with IVAs in the virtual environment can be accomplished through speech, gesture, and a combination of both modalities.

Interacting Through Speech

Two speech recognizers and a semantic text analyzer are used in the checkpoint security training system. For controlling the system itself (e.g., "*command: scenario reset*"), commands spoken directly to the user's avatar (e.g., "*command: go prone*"), and for recognition of Arabic phrases spoken by an English speaker (e.g., "*salamu alaykum*") we use a commercial finite state recognizer (Boisen, et al., 1989). Like any finite-state-automata (FSA) based recognition system, the meaning of the utterance is encoded as part of the state path that the recognizer followed, and a token representing the meaning of the utterance is passed to the behavioral rule engine.

With the exception of Arabic phrase recognition, all verbal communication with IVAs uses a speech recognizer developed as part of a real-time, two-way, speech-to-speech language translation system developed under DARPA's TRANSTAC project (Stallard, et al., 2007). While it supports English to Iraqi and Iraqi to English, as well as other languages, we currently use only its English recognition capabilities. The language model-driven, statistical recognizer was trained on a large corpus of utterances from the force protection domain (including checkpoint security), which influenced our own choice of task.

Being a statistical recognizer, it only produces a transcript. We construct the meaning by passing the text of the transcript through a semantic analysis system, Sparser (McDonald, 1993). We developed a semantic grammar of the actions and objects that come up in checkpoint activities (e.g., "*Open the trunk.*", "*Who is in the car with you?*"). If we get a complete interpretation of the transcript, we pass the semantic content of the parse to the behavioral rule engine as an intentional description.

We try each recognizer in sequence, taking the results of the first one that produces a successful result. We try the FSA-based recognizer first because it runs very fast. Since it is FSA-based, however, it will always return an interpretation, hallucinating a match against the paths in its network. To counter this, we specify a high confidence threshold on the N-Best possibilities that it returns. Next we run the utterance through the statistically-based recognizer which also returns an N-Best list of transcripts. We pass each transcript through the semantic analyzer's checkpoint grammar and take the first one where the analysis accounts for all of the words in the transcript. Word recognition in the domain is very good – its first choice is usually correct. If the behavioral rule engine cannot process output from the

semantic analyzer we try the next transcript from the recognizer's N-Best list, trying, at most, the top five transcripts from the N-Best list. If the behavioral rule engine is unable to make sense of the semantic parse, we then pass the transcript itself to the behavioral rule engine where topic-spotting heuristics are used to make sense of what was said.

Interacting Through Gesture

In our architecture, gesture perception is broken in to three distinct phases: scene perception, critical feature analysis, and gesture recognition.

For scene perception, we utilize a single TOF camera, which measures the amount of time for infrared light to travel from the camera to an object and back, producing a depth value for each pixel in the image frame. For critical feature analysis, we use a commercial software platform to extract base features about the user such as hand, head, shoulders and torso positions.

Both static and dynamic gestures are recognized. Static gestures (e.g., *stop* – arm extended in front of the body with palm out) are those that require only data on the position and alignment of various body parts at a single point in time. Dynamic gestures are those that involve changes in the body parts' positions and alignments over time (e.g., *come forward* – arm in front waving toward oneself, beckoning the subject).

A gesture is recognized as having occurred only if a set of geometrically based conditions are met. For example, in the case of the "stop" gesture, the user's hand must be out in front of their body, it must not be moving, and this stance must be held for a certain number of video frames. Once a gesture is recognized, it is directed toward any appropriate IVAs using agent perceptual filters.

Intelligent Behavior in IVAs

To achieve realistic behavior, IVAs employ filters limiting their actions and perceptions. Gestures by the user can only be seen if the user's avatar is within the field of view of the IVA, and speech from the user can only be heard if the IVA is within a predefined hearing range of the user's avatar. Speech and gesture events are dispatched asynchronously to each IVA able to perceive the communication event.

To determine the proper recipient of a communication event, IVAs employ an algorithm that combines the location and orientation of the user's avatar with the location of the IVAs in the scene. All IVAs located within the predefined hearing radius and also within a predefined field-of-view from the orientation of the user's avatar are considered viable candidates for a communication event. When a user issues a communication event, the algorithm selects from the filtered list, the avatar that satisfies the requirements of being both the closest IVA to the user's avatar as well as the closest IVA to the orientation of the avatar, with a smaller orientation delta taking precedence over proximity. This approach enables IVAs to exhibit natural reactions based on whether or not they are expected to react to the communication. If a principle intended recipient exists, but that IVA is unable to make sense of the communication, a verbal response from the intended recipient indicating a lack of comprehension is generated.

Gestures are disseminated using both *push* and *pull* mechanisms. Hand and body gesture events are detected automatically and dispatched to relevant IVAs. In addition, when handling pragmatics, or ambiguous speech references such as *there* in "*please move over there,*" the system references the gesture recognition module to determine where the user was pointing at the time of the ambiguous reference. This information is then included as part of the communication event.

Virtual objects (e.g., walls and vehicles) and entities (e.g., soldiers and civilians) in the scene are known to the IVAs and are a part of their lexicon. We utilize WordNet (Miller, 1995) to allow the user to refer to these objects and entities using broader classes of synonyms (hypernyms and hyponyms). When the user makes reference to an object in a communication utterance (e.g., "*move to the wall*"), the IVA attempts to find the nearest object in the scene mapped to that referent, allowing the IVA to respond with the appropriate behavior. In addition to knowledge about the scene, IVAs are also given detailed back-stories, such as where details about their family, where they live and who they are traveling with, enabling them to answer typical checkpoint scenario questions.

Figure 3: Scene from the checkpoint security scenario from the perspective of the trainee.

IVA knowledge and behaviors are encoded using the Drools production rule system (Bali, 2009), which integrates a Rete engine (Forgy, 1982) into an object-oriented framework. IVAs can have goals, which are achieved through execution of explicit plans. A library of actions that can be accomplished in the virtual environment is instantiated in the VBS2 scripting language. This library provides low-level capabilities such as moving within the environment and interacting with objects (e.g., getting into and out of vehicles, opening and closing doors, picking up objects, etc.) that allow the IVAs to exhibit natural behaviors. As communicating agents, a critical component of the IVAs' behavior is responding to tactical questions asked by the user. We utilized a text-to-speech engine to synthesize speech. In addition to each IVA's back-story, several personalities have been developed to provide some variation in how IVAs respond to questions from the user. Personalities include adult male, respectful older male, elderly female, and insurgent.

CHECKPOINT SECURITY SCENARIO

To highlight the features of our system, we implemented a checkpoint security training scenario. A typical checkpoint security situation provides ample opportunity to utilize all of the major components of our system such as gesturing a car forward, tactical questioning of IVAs, and avatar navigation around the scene for visual security inspections. In this section, we illustrate a typical interaction between a user in the role of a security soldier, and an IVA representing the driver of a vehicle passing through the checkpoint. Figure 3 shows a scene from an interaction between the user and an IVA. During a typical training session, a user might expect to encounter 5 or 6 vehicles, each with one or more IVAs that exhibit different behaviors and background stories.

Sample Interaction

The user approaches a projected scene displaying a checkpoint scenario, there are barriers on each side of the road and at the far end of the checkpoint, a queue of cars are waiting to be summoned forward. Facing the lead car in the queue and using a come

forward gesture, the user calls the car forward. When the car is in front of the user, he performs a stop gesture, halting the car and then switches to voice interaction.

User: "Everyone please step out of the vehicle."

The car doors open and all of the IVAs exit the car and face the speaker. The user greets the head of the household with a common Iraqi greeting.

User: "Salamu alaykum."

IVA: "Wa alaykum issalam."

User: "Please tell your family to step over by the red barrel."

The two family member IVAs walk around the car towards the barrier and stand next to a red barrel located within a designated holding area.

User: "Come over here please."

The head-of-the-household IVA approaches the user. Using the Wii remote, the user triggers a search of the IVA. Upon completion, a dialog displays the contents of the search. After acknowledging the search, the user continues to question the IVA.

User: "What's your name?"

IVA: "My name is Samir Abdul Hak."

User: "Who's with you?"

IVA: "My family, Rita and Akbar."

User: "Where do you live?"

IVA: "I'm from Baghdad."

User: "Where are you going?"

IVA: "I am going to Samarra."

User: "What's your occupation?"

IVA: "Truck Driver."

User: "Who's your employer?"

IVA: "Iraqi Transportation Network."

User: "Is this your car?"

IVA: "Yes."

User: "Do you have any weapons, ammunition, or drugs in the car?"

IVA: "No."

User: "Please open all the doors, the hood, and the trunk."

The IVA walks back to the car, performs the request, and then stands next to the car, watching the user.

User: "Please stand over there."

The user points over to the designated holding area. The IVA then walks over to the location pointed at by the user. The user walks to the vehicle and inspects it for illegal items. Once the user is satisfied with the search, he continues questioning the IVAs.

User: "Did you notice any unusual activity on your way here?"

IVA: "No."

User: "You and your family may go."

The IVAs walk towards the car, get in, and proceed through the checkpoint. The user is then free to summon the next vehicle, continuing the checkpoint operation.

CONCLUSION AND FUTURE WORK

An early version of this system was demonstrated at the 2009 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC), and an informal evaluation of performance at the conference reveals that, in general, people were easily able to use the system after a brief introduction. Speech recognition accuracy was good, with most response failures due to communications occurring outside the language and behavioral models constructed for the IVAs, rather than failures of speech recognition itself. Despite relatively high levels of ambient noise on tradeshow floor, noise was not a problem due to the use of a noise-canceling head-set microphone and the "push-to-talk" button on a Wii remote. However, on a number of occasions, users would forget to push the "push-to-talk" button. Recognition of gestures was accurate, but occasionally impaired by dark clothing. Additionally, we discovered that some individuals gestured in a different manner than expected, causing some difficulties in recognition.

While deterministic gesture recognition sufficed for this prototype system, we plan to investigate more robust gesture recognition, including probabilistic or hybrid recognition approaches, such as hidden Markov models (Wilson & Bobick, 1999) or probabilistic rule systems (Goodman, et al., 2008). It is assumed that these approaches, especially those that are temporal in nature, will better recognize dynamic gestures. In addition, future work will include improving multimodal fusion between speech and gesture inputs. While the system currently supports pointing while giving verbal orders that contain location-identifying words, it would be useful to extend this functionality to include other "free variable" words, such as pronouns; (e.g., the user could tell an IVA to move toward "her" and point to a female IVA). Additionally, the use of speech and gesture modalities to aid their respective recognitions, through co-adaptation (e.g, Christoudias, et al., 2006), would improve system performance in a number of situations. For example, saying "*come here*", increases the likelihood that a gesture being made is the "*come forward*" gesture, and vice versa.

This cross-modal recognition could also be extended to an iterative bootstrapping algorithm, at least in the case where it is reasonable to expect the gesture and the speech to be conveying redundant information. Finally, we plan to extend the language and behavior models to increase robustness and to accept a wider range of tactical questions that might be generated by a user.

REFERENCES

Bali, M. (2009). Drools JBoss Rules 5.0 Developer's Guide, Packt Publishing

Boisen S., Chow Y., Haas A., Ingria R., Roukos S., & Stallard D. (1989). The BBN Spoken Language System. Proceedings of Speech & Natural Language, 106–111, Philadelphia.

Bolt, R. A. (1980). "Put-that-there": Voice and gesture at the graphics interface. In Proceedings of the 7th annual conference on Computer graphics and interactive techniques (pp. 262-270). ACM.

Chai, J., Pan, S., Zhou, M.X., & Houck, K. (2002). Context-based multimodal input understanding in conversational systems. Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces, 87–92.

Christoudias, C.M., Saenko, K., Morency, L.P., & Darrell, T. (2006). Co-adaptation of audio-visual speech and gesture classifiers. Proceedings of the 8th International Conference on Multimodal Interfaces, 84–91.

Demirdjian, D., Ko, T., & Darrell, T. (2005). Untethered gesture acquisition and recognition for virtual world manipulation. Virtual Reality, 8(4), 222-230.

Forgy, C. (1982). Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem, Artificial Intelligence, 19, 1–37.

Goodman, N.D., Mansinghka, V.K., Roy, D., Bonawitz, K. & Tenenbaum, J.B. (2008). Church: a language for generative models. Uncertainty in Artificial Intelligence 22 (2008) 220-229

Jacob, R.J.K., Girouard, A, Hirshfield, L.M., Horn, M.S., Shaer, O., Solovey, E.T., & Zigelbaum, J. (2008) Reality-Based Interaction: A Framework for Post-WIMP Interfaces. Proceedings of the Twenty-sixth annual SIGCHI conference on Human factors in computing systems. pp. 201-210.

Maes, P., Darrell, T., Blumberg, B., & Pentland, A. (1997). The ALIVE system: Wireless, full-body interaction with autonomous agents. Multimedia Systems, 5(2), 105-112.

McDonald D. (1993). The Interplay of Syntactic and Semantic Node Labels in Partial Parsing. In: Proceedings of the Third International Workshop on Parsing Technologies, Tilburg, The Netherlands, pp. 171–186.

Miller, G.A. (1995). WordNet: A Lexical Database for English. Communications of the ACM, 38, 39-41

Stallard, D., Choi, F., Kao, C. Krstovski, K., Natarajan, P., Prasad, R., Saleem, S., & Subramanian, K. (2007). The BBN 2007 Displayless English/Iraqi Speech-to-Speech Translation System, Proceedings of INTERSPEECH 2007, 2817–2820, Antwerp, Belgium.

Wilson, A.D., & Bobick, A.F. (1999). Parametric Hidden Markov Models for Gesture Recognition. IEEE Transactions Pattern Analysis and Machine Intelligence 21, 884–900.