12

An Augmented Reality Human-Robot Collaboration System

Scott A. Green*, ** J. Geoffrey Chase* XiaoQi Chen* and Mark Billinghurst**

**University of Canterbury, Department of Mechanical Engineering*
Christchurch, New Zealand
***Human Interface Technology Laboratory New Zealand (HITLab NZ)*
University of Canterbury
Christchurch, New Zealand

1. Introduction

Interface design for Human-Robot Interaction (HRI) will soon become one of the toughest challenges that the field of robotics faces (Thrun 2004). As HRI interfaces mature it will become more common for humans and robots to work together in a collaborative manner. Although robotics is well established as a research field, there has been relatively little work on human-robot collaboration.

There are many application domains that would benefit from effective human-robot collaborative interaction. For example, in space exploration, recent research has pointed out that to reduce human workload, costs, fatigue driven errors and risks, intelligent robotic systems will need to be a significant part of mission design (Fong and Nourbakhsh 2005). Fong and Nourbakhsh also observe that scant attention has been paid to joint human-robot teams, and that making human-robot collaboration natural and efficient is crucial to future space exploration. Effective human–robot collaboration will also be required for terrestrial applications such as Urban Search and Rescue (US&R) and tasks completed robotically in hazardous environments, such as removal of nuclear waste.

There is a need for research on different types of HRI systems. This chapter reports on the development of the Augmented Reality Human-Robot Collaboration (AR-HRC) system (Green, Billinghurst et al. 2008). Fundamentally, this system enables humans to communicate with robotic systems in a natural manner through spoken dialog and gesture interaction, using Augmented Reality technology for visual feedback. This approach is in contrast to the typical reliance on a narrow communication link.

Truly effective collaboration among any group can take place only when the participants are able to communicate in a natural and effective manner. Communicating in a natural manner for humans typically means using a combination of speech, gesture and non-verbal cues such as gaze. Grounding, the common understanding between conversational participants (Clark and Brennan 1991), shared spatial referencing and spatial awareness are well-known crucial components of communication and therefore collaboration.

In a collaborative team effort it is also important to capitalize on the strengths of each member of the team. For example, humans are good at problem solving and dealing with unexpected events, while robots are good at repeated physical tasks and working in hazardous environments. An effective human-robot collaboration system should therefore exploit these strengths. The AR-HRC system does just that by enabling the human and robot to discuss a plan, review it and make adjustments to it. It then lets the robot execute the plan with interaction from the human team member, if or as, warranted. If an unexpected situation arises the robot can discuss the problem with its human partner and arrive at a solution agreeable to both, similar to the collaborative behaviour in human teams.

Augmented Reality (AR) is a technology for overlaying three-dimensional virtual graphics onto the users view of the real world (Azuma 1997). AR allows real time interaction with these virtual graphics, enabling a user to reach into the augmented world and manipulate it directly. AR is used in this research to provide a common 3D graphic of the robot's workspace that both the human and robot reference.

The internal state of the robot and its intended actions are displayed through the virtual imagery in the AR environment. The human team member is thus able to maintain situational awareness of the robot and its surroundings, thereby giving the human-robot team the ability to ground their communication. By coupling AR with spoken dialog a multi-modal interface has been developed that enables natural and efficient communication between the human and robot team members, thus enabling effective collaboration.

In this chapter, the development of the Augmented Reality Human-Robot Collaboration (AR-HRC) system is discussed. Related work is reviewed and its influence on the design of the AR-HRC system is discussed. AR is introduced, and its benefits in terms of HRC are reviewed. The initial development of an AR multi-modal interface is discussed and then the architectural design of the AR-HRC system is presented. A case study incorporating a mobile robot into the AR-HRC system is then presented and the robustness and effectiveness of this system is evaluated in a performance experiment. The chapter ends with a set of summary conclusions and a presentation of future research directions.

2. Related Work

2.1 Human-Robot Interaction

In this work, collaboration is defined as "working jointly with others or together especially in an intellectual endeavor". Clark and Brennan provide a communication model to interpret collaboration (Clark and Brennan 1991). In this model, conversation participants attempt to reach shared understanding or common ground. Common ground refers to the set of mutual knowledge, shared beliefs and assumptions that collaborators have. Therefore, an effective human-robot collaborative team needs to be able to easily reach common ground.

Milgram *et al* (Milgram, Zhai et al. 1993) highlighted the need for combining the attributes that humans are good at with those that robots are good at to produce an optimized human-robot team. Milgram *et al* pointed out the need for HRI systems that can transfer the interaction mechanisms that are considered natural for human communication to the precision required for machine information.

Adjustable autonomy, enabling the system to vary the level of robot autonomy, increases performance and is an essential component of an effective human-robot collaborative system. The ability to vary the level of robot autonomy has been seen to improve performance (Tsoukalas and Bargiotas 1996; Ishikawa and Suzuki 1997; Bechar and Edan 2003). Varying the level of autonomy of human-robotic systems allows the strengths of both the robot and the human to be maximized. It also enables the system to optimize the problem solving skills of a human and effectively balance that with the speed and physical dexterity of a robotic system.

By giving the robotic system the ability to ask help of its human counterpart, performance is improved through the addition of human skills, perception and cognition. The system also benefits from human advice and expertise (Fong, Thorpe et al. 2002). Adjustable autonomy enables the robotic system to better cope with unexpected events such as being able to ask its human team member for help when necessary. In essence, adjustable autonomy gives the robot the ability to act as a collaborative partner.

Situational awareness, or being aware of what is happening in the robot's workspace, is also essential in a collaborative effort. The lack of situational awareness decreases the performance of human-robot interaction (Murphy 2004; Yanco, Drury et al. 2004). In more extreme cases, the lack of situational awareness can result in a dangerous collision as was found by Ellis (Ellis 2000). In particular, when the unmanned space ship Progress collided with Mir, Ellis found that the lack of situational awareness was one of the key cognitive factors that caused the collision. Thus, improving situational awareness in human-robot interaction also improves performance (Scholtz, Antonishek et al. 2005).

The design of an effective human-robot collaboration system must afford the use of natural speech (Nourbakhsh, Bobenage et al. 1999; Kanda, Ishiguro et al. 2002). However, speech alone is not enough to complete the grounding process. Therefore, a multi-modal approach must be taken (Huttenrauch, Green et al. 2004; Sidner and Lee 2005) which will enable the human-robot team to communicate in a similar manner to that of a human-human team. According to Clark and Brennan (Clark and Brennan 1991) the process of grounding involves communication using a range of modalities, including voice, gesture, facial expression and non-verbal body language.

Fong *et al* (Fong, Kunz et al. 2006) note that for humans and robots to work together as peers, the human-robot system must provide mechanisms for both humans and robots to communicate effectively. For example, to enable the use of spatial dialog, it is an inherent requirement that common frames of references be used (Skubic, Perzanowski et al. 2004). A robot should reach a common understanding in communication by understanding the conversational cues used by humans, such as gaze direction, pointing and gestures.

2.2 Augmented Reality

Augmented Reality (AR) has many benefits that help to create a more ideal environment for human-robot collaboration (Green, Billinghurst et al. 2008). In a typical AR set-up, the user wears a head mounted display (HMD) with a camera mounted on it. The output from the camera is fed into a computer, augmented with 3D graphics and then fed back into the HMD. The user sees an enhanced view of the real world through the video image in the HMD, as shown in Figure 1. Therefore, AR blends virtual 3D graphics with the real world in real time (Azuma 1997). This type of AR set-up is commonly called a video-see-through AR interface.

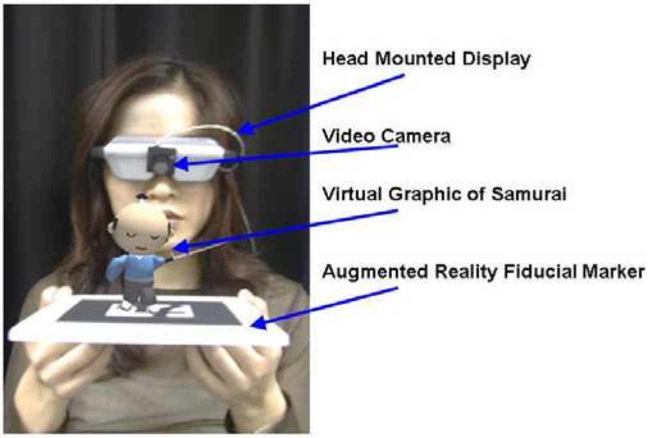

Fig. 1. Video-see-through AR interface (Billinghurst, Poupyrev et al. 2000).

Square fiducial patterns are placed in the real environment with a unique symbol in the middle of each pattern. Computer vision techniques are then used to identify the unique symbols, calculate the camera position and orientation from these symbols, and display 3D virtual images aligned with the position of the fiducial patterns (ARToolKit 2008). This augmented view is then fed into the HMD providing the user with a seamless combination of the real world view and virtual graphics, where the virtual images appear fixed to the fiducial patterns. The ability to manipulate the physical markers with fiducial patterns on them enables direct real-time interaction with the 3D virtual content.

Multiple users can view the same fiducial patterns and therefore have their own perspective of the 3D virtual content. Since the users see each other's facial expressions, gestures and body language, AR therefore supports natural face-to-face communication. This interaction demonstrates that a 3D collaborative environment enhanced with AR content can seamlessly enhance face-to-face communication and allow users to naturally work together (Billinghurst, Poupyrev et al. 2000; Billinghurst, Kato et al. 2001). Shared visual workspaces of this type have been shown to enhance collaboration as they increase situational awareness (Fussell, Setlock et al. 2003).

AR also improves collaboration by allowing the use of physical tangible objects for ubiquitous computer interaction. Thus making the collaborative environment natural and effective by allowing participants to use objects for interaction that they would normally use in a collaborative effort (Billinghurst, Grasset et al. 2005). AR provides rich spatial cues permitting users to interact freely in space, supporting the use of natural spatial dialog (Billinghurst, Poupyrev et al. 2000).

Bowen *et al* (Bowen, Maida et al. 2004) showed through user studies that the use of AR resulted in significant improvements in robotic control performance. Similarly, Drury *et al* (Drury, Richer et al. 2006) found that for operators of Unmanned Aerial Vehicles (UAVs) augmenting real-time video with pre-loaded map terrain data resulted in a significant difference in the comprehension of 3D spatial relationships compared to 2D video alone. The AR interface provided better situational awareness of the activities of the UAV.

Providing the human with an exo-centric view of the robot and its surroundings enables the human to maintain situational awareness of the robot and gives the human-robot team the ability to ground their communication. AR can thus provide a 3D world, within which both the human and robotic system can operate, a shared space (Billinghurst, Poupyrev et al. 2000). This use of a common 3D world provides common reference frames for both the human and robot.

AR supports the use of spatial dialog and deictic gestures, allows for adjustable autonomy, supports multiple human users, and allows the robot to visually communicate to its human collaborators its internal state through the use of graphic overlays on the real world view of the human. AR also enables a user to experience a tangible user interface, where physical objects can be manipulated to affect changes in the shared 3D scene (Billinghurst, Grasset et al. 2005), thus allowing a human to reach into the 3D world of the robotic system and manipulate it in a way the robotic system can understand. Therefore, by taking advantage of the benefits of AR, a robust human-robot collaboration system can be more effectively created.

3. Multi-modal Augmented Reality Interaction

To aid in the development of a multi-modal interaction framework required for the AR-HRC system the first step was to investigate previous multi-modal research and create a multi-modal AR application. One of the first interfaces to support speech and gesture recognition in a multi-modal interface was the Media Room (Bolt 1980). The Media Room allowed the user to interact with a computer through voice, gesture and gaze. Bolt's work showed that gestures combined with natural speech (multi-modal interaction) lead to a powerful and more natural human machine interface.

Speech and gesture compliment each other and when used together create an interface more powerful than either modality alone. The difficulty of using speech alone was demonstrated by Kay (Kay 1993) who constructed a speech driven interface for a drawing program. Even simple cursor movements around the screen required a time consuming combination of continuous and discrete vocal commands.

Tangible User Interfaces (TUIs) use real-world objects as the interaction devices for a software program or computer (Ishii and Ullmer 1997). Therefore, TUIs are extremely intuitive to use because physical object manipulations are mapped one-to-one to virtual object operations (Fitzmaurice and Buxton 1997). Another benefit of a TUI is that it naturally supports sharing and collaboration.

TUIs are a viable approach for interaction with AR applications as they enable users to interact naturally by manipulating real world objects. Thus, the principles of TUIs can be combined with AR's display capabilities in an interface metaphor known as Tangible Augmented Reality (TAR) (Kato, Billinghurst et al. 2001). A TAR interface supports the presentation of 3D virtual objects anywhere in the physical environment, while simultaneously allowing users to interact with this virtual content using real world physical objects (Kato, Billinghurst et al. 2000). An ideal TAR interface facilitates seamless display and interaction, removing the functional and cognitive seams found in traditional AR and TUI interfaces.

To investigate these types of multi-modal interaction in an AR environment, a system was developed that used speech and paddle based gestures as input to an AR system. The system is a modified version of the VOMAR application for tangible manipulation of virtual furniture in an AR setting (Kato, Billinghurst et al. 2000). The VOMAR application used a single modality with a paddle input device that allowed the user to perform different tasks by using real world paddle movements to manipulate virtual objects in an AR setting.

The objective of the Multi-modal AR System (MARS) was to allow people to easily and effectively arrange AR content using a natural mixture of speech and gesture input. A single fiducial marker located on the end of the paddle allows the ARToolKit library to efficiently locate the paddle's position and orientation in the AR environment. A4 sized pages containing an array of fiducial markers serve as menu pages holding virtual furniture models, see Figure 2. A separate sheet also containing AR fiducial markers serves as the workspace. This page displays a 3D graphic of an empty room where the virtual models of furniture are to be placed. The square fiducial markers printed on these pages are used by the ARToolKit library to locate and place the virtual content.

Furniture, for example, can be arranged in a room by selecting various pieces of virtual furniture and placing them in the virtual room, as seen in Figure 3. As the user looks at each of the A4 container pages through a Head Mounted Display (HMD), they see different sets of virtual furniture. The 3D virtual models appear superimposed over the real pages aligned with the fiducial markers. Looking at the workspace page for the first time the user sees an empty virtual room. The user is able to transfer objects from the menu pages to the virtual room using paddle and speech commands.

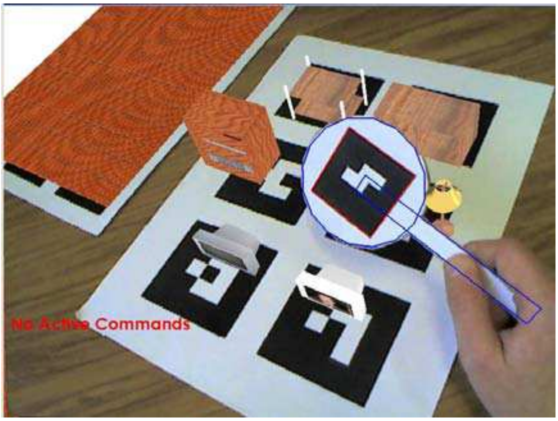

Fig. 2. An A4 sized sheet with an array of square fiducial markers is used as a menu page containing various pieces of furniture. The user can be seen holding the real world paddle that is used to interact with the virtual content. This is the view seen in the user's HMD.

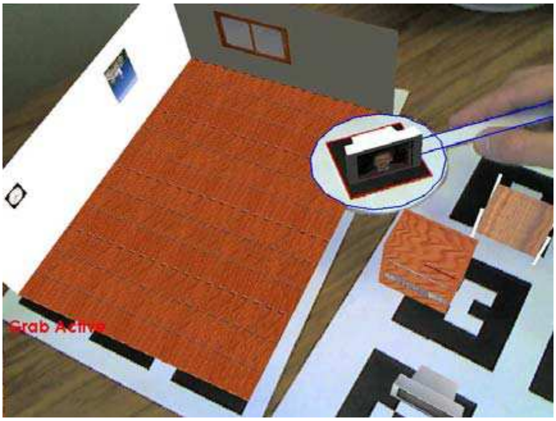

Fig. 3. A separate sheet containing virtual markers locates the virtual room. Shown here is the view the user sees with the room initially empty and a virtual object ready to place in the room.

The user can also modify the position and orientation of furniture already in the virtual room through the use of the real world paddle and speech input. The system provides visual and audio feedback to the user. The speech interpretation result is shown on the screen and audio feedback is provided after the speech and paddle gesture command. Therefore, the user is immediately notified when the system recognizes a speech or gesture. The MARS architecture is shown in Figure 4. The speech processing module recognizes the spoken dialog of the user and is also responsible for the text to speech (TTS) of the application. The Dialog Management System (DMS) compares the spoken dialog that the speech processor recognized and parsed with predefined goals for the system. The speech processor and DMS make use of the Ariadne spoken dialog system (Denecke 2002).

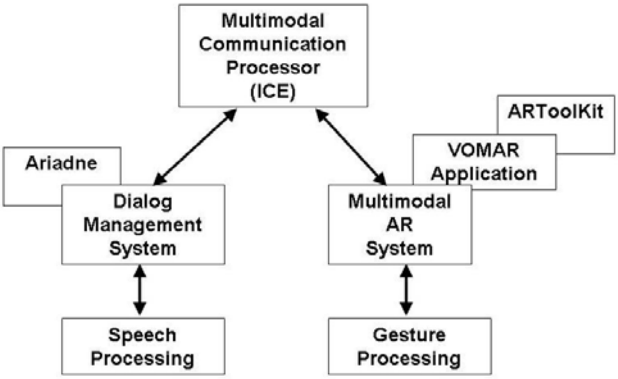

Fig. 4. Architecture for the MARS application.

When a goal has been reached through spoken dialog, the DMS sends the appropriate command to the MARS module via the Multi-modal Communication Processor (MCP). The MCP is built upon the Internet Communications Engine (ICE) (ZeroC 2008). If the command sent is to be combined with a gesture, the MARS module checks for the paddle to be in the users view and calculates its position. At the same time the MARS module calculates the location of all virtual objects, whether they are on the menu pages or in the virtual room, and compares these locations to that of the paddle. When the paddle is within a certain distance of a virtual piece of furniture, this piece of furniture becomes active and the verbal command sent in from the DMS is applied to the selected piece of furniture. The MARS module is written in C++ and uses the VOMAR and ARToolKit (ARToolKit 2008) libraries.

Thus, a user may combine speech commands and paddle gestures to interact with the system. To understand the combined speech and gesture, the system must fuse input from both streams into a single understandable command. This fusion is achieved by recording the time each input occurs, as given by an event time stamp. The paddle and speech input can therefore be considered for fusion only if the input time stamps from both input streams are within a certain time window of each other.

The multi-modal AR application is capable of working in three different modes, a gesture only mode, speech input with static paddle placement, and full paddle gesture combined with speech. The gesture only mode works essentially the same as the initial VOMAR application. In this mode, the user interacts with the system through paddle commands only and as a result no explicit fusion strategy is needed.

The gesture only mode consists of a variety of interaction techniques. The user is able to pick an object from the menu page with an empty paddle by holding the paddle close to the object. If the paddle is empty and placed on a virtual object, the object is picked up after the paddle has remained in this position for a short period of time.

If there is an object attached to the paddle, when the paddle is tilted the object will slide off the paddle and into the empty virtual room. If there is an object attached to the paddle, the object will be deleted from the paddle when the paddle is shaken from left to right. An object already placed in the room can be moved around by “pushing” it with the paddle. An item of furniture that is placed in the room can be deleted by hitting it with the paddle.

Once an object is on the paddle, it can be picked up and viewed from any viewpoint. These interactions are very natural to perform with the real paddle, so in a short period of time a user can assemble a fairly complex arrangement of virtual furniture. However, placement of the virtual furniture is this manner is not very precise.

A second mode of interaction for the system is to use speech combined with static paddle placement. In this interaction mode the user interacts with the virtual content using speech and paddle placement. However, the system only considers the static paddle pose at a particular time and fuses this information with the speech recognition result to interpret the combined speech and gesture commands.

This mode works as follows, when a speech command is recognized it is checked against a set of goals. If a match is found, the appropriate command id number is sent to the AR application. For example, consider the speech input "grab this" while the user has placed the paddle in the proximity of a virtual object on one of the menu pages. The system will check the paddle position, if the paddle is close enough to the object, the object will be grabbed, or selected from the contents page and placed on the paddle for further action. If the paddle position is not close enough, the object will not be grabbed. However, the grab command

remains active for five seconds. So, the user can move the paddle closer to the object. If the paddle is moved to within the proximity limit and the difference between the current time stamp and previous speech input time stamp is five seconds or less, the object will be “grabbed”, otherwise the user has to repeat the speech command.
A list of speech commands the system can process are given below:

Delete Command:
This command will delete an object from the paddle or from the workspace. If there is an object on the paddle, it will be deleted. If there is no object on the paddle and the workspace is in view, the object the paddle is touching will be deleted.

Translate Command:
If the workspace is in view, this command attaches a virtual object in the workspace to the paddle so that it follows the paddle translation. The object will be released from the paddle after the user gives the *Stop* or *Place* command.

Rotate Command:
This command has a similar function as the Translate command. It attaches a virtual object in the workspace to the paddle so that it can follow the paddle rotation. The object will be released from the paddle after the user gives the *Stop* or *Place* command.

Move Command:
This command combines the Translate and Rotate commands. It attaches a virtual object from the workspace to the paddle so that it can follow both the paddle translation and rotation. The object will be released from the paddle after the user gives the *Stop* or *Place* command.

Place Command:
If there is an object attached to the paddle, this command places the attached object at the paddle location in the workspace.

Stop Command:
This resets a *Delete*, *Translate*, *Rotate* or *Move* command.

The final mode of interaction is full paddle gesture combined with speech. The user is able to interact with the system using both speech and continuous paddle gestures. This mode is a combination of the two modes explained previously. For example, the user can give a speech command "grab this" and then place the object using the paddle tilting gesture. In this manner, the user can easily combine speech and paddle gesture input and choose which interaction technique is more appropriate for the given task.
A user study was conducted to determine if the multi-modal interface actually improved the efficiency of user interaction in an AR environment. The results of this study did indeed show that a multi-modal interface improved performance for given tasks in an AR environment (Irawati, Green et al. 2006). These positive study outcomes provided the impetus to incorporate this type of interaction into the AR-HRC system.

4. AR-HRC System Architectural Design

A multi-modal approach has been taken for the architectural design of the AR-HRC system. This architecture combines speech and gesture through the use of AR that allows humans to naturally communicate with robotic systems. Through this architecture the robotic system receives the discrete information it needs to operate while allowing human team members to communicate in a natural and effective manner by referencing objects, positions, and intentions through natural gesture and speech. The human and the robotic system each maintain situational awareness by referencing the same shared 3D visuals of the workspace in the AR environment.

The architectural design is shown in Figure 5. The speech-processing module recognizes human speech and parses this speech into the appropriate dialog components. When a defined dialog goal is achieved through speech recognition, the required information is sent to the Multi-modal Communication Processor (MCP). The speech-processing module also takes information from the MCP and the robotic system and synthesizes this speech for effective dialog with human team members. As opposed to the earlier approach using a third party dialog system, it was decided to create a speech processing module catered to the specific needs of the AR-HRC. This speech processing module is based upon the Microsoft Speech SAPI 5.1 (MicrosoftSpeech 2007).

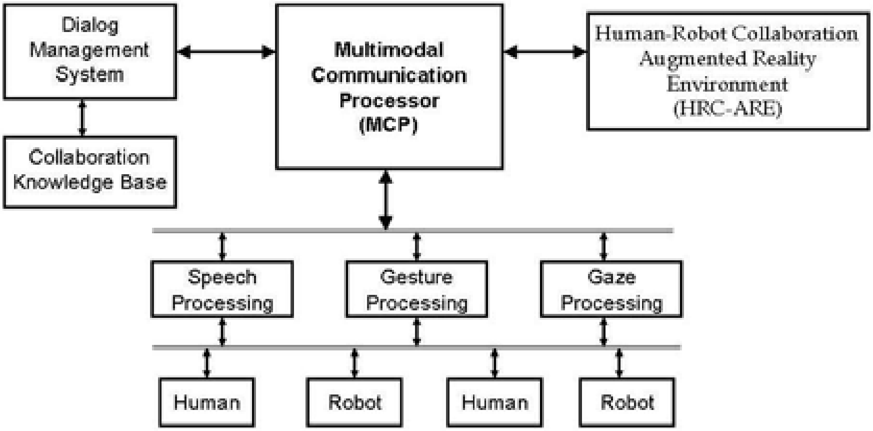
Dialog Management System
Collaboration Knowledge Base
Multimodal Communication Processor (MCP)
Human-Robot Collaboration Augmented Reality Environment (HRC-ARE)
Speech Processing
Gesture Processing
Gaze Processing
Human
Robot
Human
Robot

Fig. 5. The AR-HRC system architecture.

Gesture processing enables the human team members to use deictic referencing and normal gestures to communicate effectively with a robotic system. It is imperative that the system be able to translate the generic references that humans use, such as pointing into 3D space and saying “go here”, into the discrete information a robotic system needs to operate. The gesture-processing module recognizes human gestures and passes this information to the MCP.

The MCP combines the speech from the speech processing module, the gesture information from the gesture-processing module and uses the Human-Robot Collaboration Augmented Reality Environment (HRC-ARE) to effectively resolve ambiguous deictic references such as *here, there, this* and *that*. The HRC-ARE is built upon the OSGART libraries (Looser, Grasset et al. 2006). OSGART is an Open Scene Graph (Open Scene Graph 2008) wrapper for the ARToolKit (ARToolKit 2008) and was selected for use in the AR-HRC for its high level rapid prototyping approach to creating virtual content for AR environments.

The disambiguation of the deictic references is accomplished in the AR environment's 3D virtual replication of the robot's world. The human uses a real world paddle to reach into and interact with this 3D virtual world. This tangible interaction, using a real world paddle to interact with the virtual 3D content, is a key feature of AR that makes it an ideal platform for HRC.

The gaze-processing module interprets the users gaze through the use of a camera mounted on a head mounted display (HMD). Through the use of the computer vision algorithms in the ARToolKit, the gaze direction of the human into the 3D world can be computed. The use of individual HMDs enables multiple human team members to view the HRC-ARE from their own perspective.

This personal viewing of the workspace results in increased situational awareness. Each team member views the work environment from his or her own perspective. They can change perspective simply by moving around the 3D virtual environment as they would a real world object, or they could move the 3D virtual world and maintain their position by moving the real world fiducial marker that the 3D world is "attached" to.

Not only do the human team members maintain their perspective of the robotic system's work environment, but they are also able to smoothly switch to the robot's view of the work environment. This ability to smoothly switch between an exo-centric (God's eye) view of the work environment to an ego-centric (robotic system's) view of the work environment is another feature of AR that makes it ideal for HRC and enables the human to quickly and effectively reach common ground and maintain situational awareness with the robot.

The Dialog Management System (DMS) is tasked to be aware of the communication that needs to take place for the human and robot to collaboratively complete a task. The MCP takes information from the speech, gesture and gaze processing modules along with information generated from the HRC-ARE and supplies it to the DMS. The DMS is responsible for combining this information and comparing it to the information stored in the Collaboration Knowledge Base (CKB).

The CKB contains information pertaining to what is needed to complete the desired tasks that the human-robot team wishes to complete. The DMS then responds through the MCP to either human team members or the robotic system, whichever is appropriate, facilitating dialog and tracking when a command or request is complete.

The MCP is responsible for receiving information from the other modules in the system and sending information to the appropriate modules. The MCP is thus responsible for combining multi-modal input, registering this input into something the system can understand and then sending the required information to other system modules for action. The result of this system design is that a human is able to use natural speech and gestures to collaborate with robotic systems.

5. A Case Study

In this section, the implementation of the AR-HRC with a mobile robot is described. This implementation enables a user to communicate with a mobile robot using natural speech and gestures. An example of this type of exchange would be the human gesturing to a point in 3D space and saying "go here" or "go behind that".

The AR-HRC system takes this ambiguous information and translates it into discrete information for the robot to process and act upon. The robot then responds using speech and portraying its plans in the 3D virtual content of the AR environment. In this manner the human is able to understand the intentions and beliefs of the robot collaborative partner.

When the robot responds using speech, it randomly selects from a list of possible responses appropriate for the given situation. The situations for which the robot responds verbally are predetermined, thus when one of these situations arises, the system randomly selects a response that has been defined for that given situation. By issuing random responses the dialog of the robotic system feels more natural to the human. If the robot were to use the same phrase each time it needed to respond to a given situation, it would feel unnatural to the human and detract from the feeling of being collaborative partner.

As described previously, the use of AR technology enables the human collaborative partner to use natural speech and gestures. The AR environment also provides a common 3D spatial reference for both the robot and human, thus providing a means for grounding of communication and maintaining spatial awareness.

During implementation of the AR-HRC system with the mobile robot the gesture processing was developed to be modal. Verbal commands issued by the human determine which modality the real world paddle will be used in. The paddle can be used as a pointer enabling the human to point into the 3D virtual world of the robot and select a point or object. A second modality lets the human use the paddle for natural gestures.

Natural gestures have been defined from those used by participants in a Wizard of Oz study that was conducted to determine what kind of natural speech and gestures would be used to collaborate with a mobile robot (Green, Richardson et al. 2008). Natural gestures have been defined to communicate to the robot to move straight forward, turn in place, move forward while turning, back up and stop. At any time the user can issue a verbal command resulting in a true multi-modal experience.

The real world paddle has a fiducial marker on the end opposite the handle. The paddle is flat and has a fiducial marker on both sides. By placing markers on both sides of the paddle the vision system will be able to see the marker no matter which way the user holds the paddle. In the pointer mode a virtual pointer appears attached to the paddle. Therefore when the user moves the real world paddle around the virtual pointer follows the motion of the real world paddle.

The virtual pointer is the visual cue used by the human to select locations and objects in the virtual world. When the virtual pointer intersects other virtual content in the scene it is occluded thus providing the user with the cues necessary to determine precisely the point or object selected by the virtual pointer. But when the paddle is used for natural gestures the virtual pointer does not appear. Instead different visual indicators appear to let the user know what command they are giving. If the user holds the paddle straight out in front of them it is interpreted as a go forward gesture and an icon appears alerting the user of this.

The AR-HRC calculates the position and orientation of the fiducial marker on the paddle. The system then determines the orientation of the paddle relative to the user's point of view.

If the paddle is held straight out in front of the user then the orientation angles of the paddle will fall within certain defined threshold values indicating to the system that a move straight forward command has been issued.

Similarly, when the paddle is moved to either side of straight in front of the user the system calculates the angle from straight ahead and converts this information into a turn. To turn the robot in place the user starts from the straight up position and rotates their arm about their elbow to the right or left. To go in the reverse direction the user places the paddle in a straight up position. Any position of the paddle not specifically defined is interpreted as a stop command and is relayed to the user by displaying a stop sign on the paddle. See Figure 6. for various paddle-gesture commands.

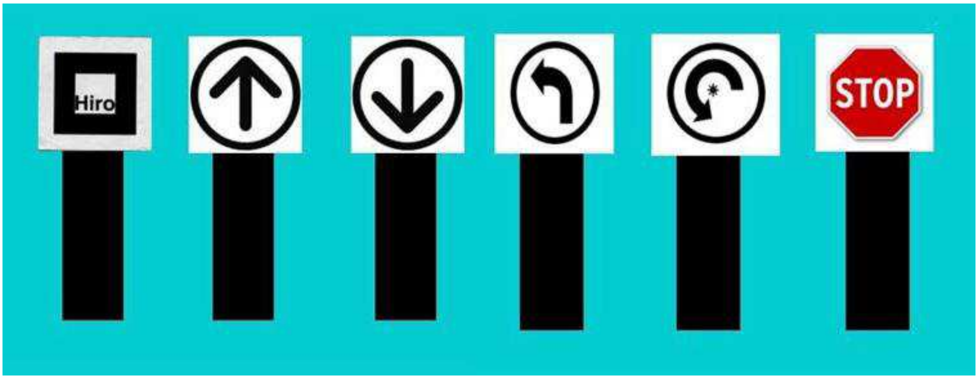

Fig. 6. The image on the far left is of the paddle as seen in the real world. The remainder of images show the AR augmented view showing, from left to right, forward, reverse, forward turn left, turn left in place and stop.

The gaze-processing module defines the gaze direction of the user through the computer vision techniques made available from the use of the ARToolKit libraries. The line of sight of the user into the virtual world is computed by calculating the position of the camera mounted on the HMD, this calculation represents the users gaze direction. By comparing this position to the position and orientation of the marker set that represents the robots virtual world the user gaze direction can be determined.

The gaze direction of the user in the AR environment is used to define spatial references such as "behind" and "to the right of" objects selected using the real world paddle. By knowing where the user is in reference to the objects in the virtual scene these spatial references can be defined in the reference frame of the user, as described in (Irawati, Green et al. 2006). This information is then converted into the reference frame of the robot. The conversion is made possible through the use of AR, which provides a common reference frame for both the robot and human collaborators. The desired location is then sent to the robot where it uses its autonomous capabilities to move to the position in the real world.

As a case study a Lego MINDSTORMS™ NXT (The Lego Group 2007) mobile robot in the Tribot configuration was used as a mobile robot to collaborate with. The NXT robot can be seen in Figure 7. To incorporate the mobile robot into the system the NXT++ libraries (NXT++ 2007) were used. These libraries represent an interface to the MINDSTORMS™ robot written in C++ that enables a PC to communicate with the robot through a Bluetooth connection. A Lego MINDSTORMS™ robot was chosen because it is a simple low cost platform to prove out the functionality of the AR-HRC system.

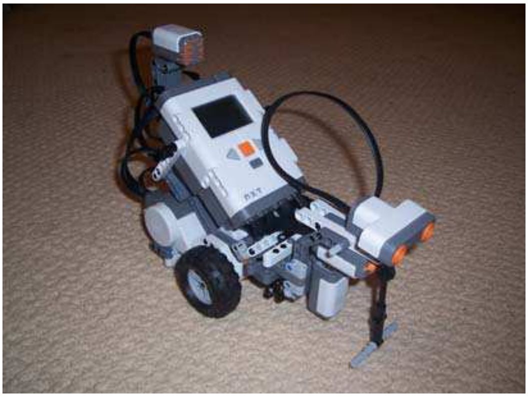

Fig 7. Lego MINDSTORMS™ NXT robot (The Lego Group 2007).

An obstacle course was created for the NXT robot to manoeuvre through. A virtual representation of this world and the NXT robot was created to be used in the AR environment. The motions of the real robot were calibrated to match those of the virtual representation of the NXT robot in AR.

The NXT robot used had one ultrasonic sensor on the front to sense objects and measure the distance to them. The robot also had a touch sensor on the front that would stop the robot if triggered to avoid colliding into objects in its world. The limited sensing ability of the robot allowed us to take advantage of dialog to ensure the robot took a safe path.

An example of using dialog to ensure safe robot motion would be when the robot had to back up. With no rear sensors the robot was unable to determine if a collision was imminent. In this case, the robot asked the human if it was ok to move in reverse without hitting objects in its environment prior to commencing movement. Once the robot received confirmation that the path in the reverse direction was clear, it began to move in the reverse direction. Since the robot had to ask for guidance to complete the reverse manoeuvre, the user was aware that the robot might need assistance. It was then assured that the user has maintained spatial awareness, which in turn enabled a collaborative human-robot exchange and the resulting safe execution of robot motion.

A heads up display was used to keep the human informed of the internal state of the robot. The human could easily see the direction the robot was moving, the battery level, motor speeds, paddle mode and server status. Figure 8 is an example of the view provided to the human through the HMD. The internal state of the robot is easily identifiable as is the robots intended path and progress.

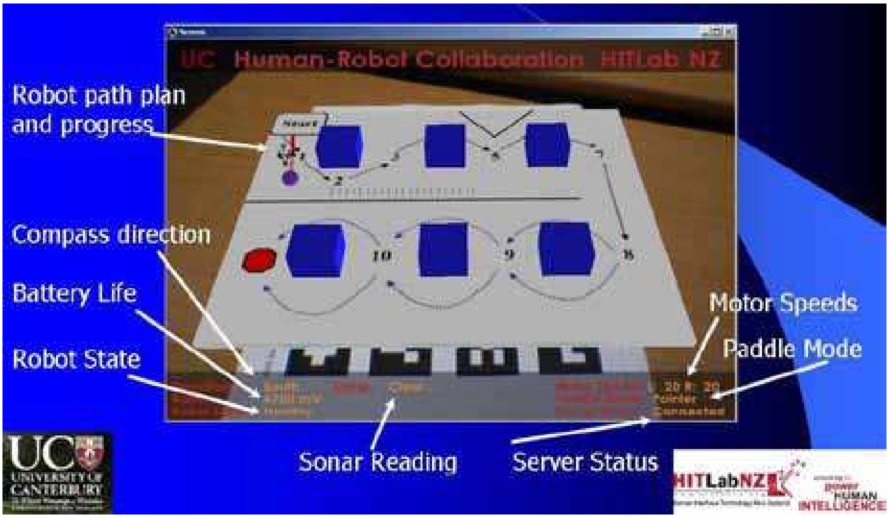

Fig. 8. View provided to user in HMD highlighting robot state information and path.

Since the robot's environment was modelled in 3D and used as the virtual scene in AR, the human is provided the means to gain a feeling of presence in the robot's world. The system allows the human to naturally communicate with the robot in the modality most comfortable to the user. Given the limitations of the MINDSTORMS™ robot sensors the human had to do more monitoring than would be necessary with a more autonomous robot. Another benefit of using AR as a means to mediate the communication between the robot and human is the ability to smoothly transition from an exo-centric (god's eye view) to an ego-centric view. This means the user can smoothly transition from a bird's eye view of the robot in its environment to the view provided by the robot's camera, and vice-versa. The user is able to issue a verbal command to switch from one viewpoint to the other. This ability to view the robot's world from two vantage points increases the situational awareness of the human by allowing the scene to be viewed from the various vantage points.

The human and robot are able to create a path plan and review this plan before the robot is set in motion. The ability to review the plan prior to execution provides the means for detection of unexpected situations and the identification of probable collisions. The path plan can be interactively created and modified through dialog and gesture interaction within the AR environment.

A path can be planned and then viewed as overlays on the 3D virtual model of the robot's environment. The user can choose to show the planned path or hide the trajectory if the overlay interferes with viewing other important parts of the environment. Waypoints can be added to or deleted from the path plan to ensure smooth motion and obstacle avoidance. The result is that the motion of the robot, once it executes its collaboratively designed plan, is smoother and collision free.

During review of a path plan the robot gives the user verbal feedback if a collision is imminent, as well as when a collision happens. This feedback allows the user to modify the plan to ensure collision free movement. This setup also enables the user to provide a margin of safety between the robot and any objects it might possibly collide with. This type of interaction between the robot and human in creating the path plan, identifying possible problems with the plan and eliminating these issues highlights the collaborative nature of the human-robot interface.

6. Performance Experiment

This section discusses an experiment that compared three user interface techniques for interaction with a simulated mobile robot that was located remotely from the user. A typical means of operating a robot in such a situation is to teleoperate the robot using visual cues from a camera that displays the robot's view of its work environment. The operator has a difficult time maintaining awareness of the robot in its surroundings due to this single ego-centric view. Therefore, this interface was compared with two versions of the AR-HRC system.

6.1 Experimental Design

The task for the user study was to work with a simulated robot to get it through a predefined maze. Three conditions were used:

- A typical teleoperation mode with a single ego-centric view from the robot's onboard camera. This condition was called the *Immersive Test*.
- A limited version of the AR-HRC system that allowed the user to see the robot in its work environment in AR and interact with the robot using speech and gesture, but without pre-planning and review of the robot's intended actions. This condition was called *Speech and Gesture no Planning (SGnoP).*
- The full AR-HRC system that allowed the human to view the robot in the AR environment, use spoken dialog and gestures to work with the robot to create a plan and review this plan prior to execution. This was the *Speech and Gesture with Planning, Review and Modification (SGwPRM)* condition.

The three conditions are, therefore, distinguished by increasing levels of collaboration or communication channels.

6.2 Participants

Ten participants were run through the experiment, seven male and three female. Ages ranged from 28 to 80 and all participants were working professionals. Six participants had Bachelor's degrees and four advanced degrees. Seven of the participants were engineers while the other three had non-scientific backgrounds. Overall the users rated themselves as not familiar with robotic systems, speech systems or AR.

6.3 Procedure

The first step of the experiment was to have each participant fill out a demographic questionnaire to evaluate their familiarity with AR, game playing experience, age, gender and educational experience. Since speech recognition was an integral part of the experiment it was necessary to have each participant run through a speech training exercise. This training created a profile for each user so that the system was better able to adapt to the speech patterns of the individual participant.

The objective of each trial was then explained to the participants. They were told that they would be interacting with a mobile robot to get it through the predefined maze shown in Figure 9. The maze contained a defined path for the robot to follow and various obstacles the robot would need to maneuver around. The black lines indicate a path that needed to be followed while the blue lines indicate that the user had the choice of how to proceed. The participants were told that the robot must arrive at each of the numbers on the map as this was going to be a measure of accuracy for the test. Other parameters measured were impending collisions, actual collisions and time to completion.

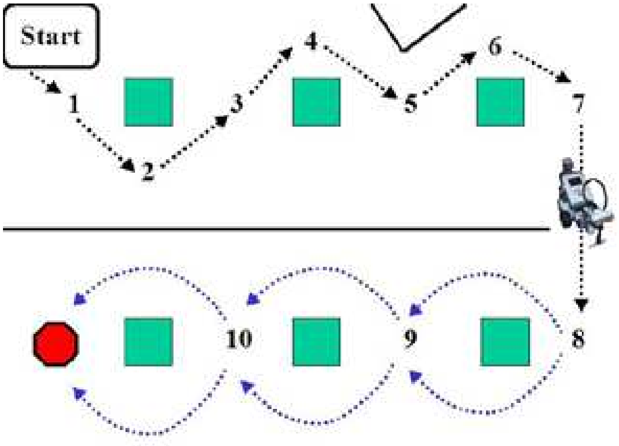

Fig. 9. User study maze, black lines indicate defined path, blue lines indicate users choice.

It was explained to the participants that the robot was located remotely. The effect on the trials this had was that when the robot was directly driven a time delay would be experienced. So a delay in reaction of the simulated robot was not the system failing, but was the result of the time taken for the commands to reach the robot and the update from the robot to arrive back to the user.

The experimental setup used was a typical video see through AR configuration. A webcam attached to an eMagin Z800 Head Mounted Display (HMD) (eMagin 2008) were both connected to a laptop PC running ARToolKit based software. Vision techniques were use to identify unique markers in the user’s view and align the 3D virtual images of the robot in its world to these markers. This augmented view was presented to the user in the HMD. Figure 10 shows a participant using the AR-HRC system during the experiment.