2011 IEEE International Conference on Robotics and Automation
Shanghai International Conference Center
May 9-13, 2011, Shanghai, China

# Comparison between Two Mixed Reality Environments as a Teleoperation Interface

Ida Bagus Kerthyayana Manuaba, Ken Taylor and Tom Gedeon

***Abstract*— An important aspect of teleoperation is situational awareness through visualization. The actual operation and control of a remote machine must be supported by an interface which provides enough information through visualization from a remote location to complete a task. This can be achieved with a Mixed Reality (MR) environment. The concept is to combine information from the real world and a virtual world. An experiment was conducted to assess the differences between two platforms and to determine interface features required to maximize operator performance and satisfaction. The result indicates that both mixed reality environments tested were suitable for teleoperation where sufficient information to perform the task could be modeled in the virtual world. However, one of the environments turned out to be superior where the task required information in the video but not modeled in the virtual environment. The preferred environment provided overlays on the video that were updated live as the model was manipulated where the other environment updated video overlays on completion of the manipulation.**

## I. Introduction

THIS paper presents an experiment designed to investigate the impact of combining virtual reality and information from the real world within a mixed reality environment for teleoperation. Monferrer and Bonyuet [1] state "with virtual reality, one is able to figure out how to view a problem and complete a task in multiple ways". However, a virtual environment built to simulate the real world is always incomplete. An alternative approach is a mixed reality environment. According to Milgram [2], mixed reality is a representation of the real and virtual world objects which are presented together within a single display and it aims to link the virtual entities with the real world.

Manuscript received September 15, 2010. This work was supported in part by the Commonwealth Science Industrial and Research Organization (CSIRO) ICT Center and School of Computer Science, College of Engineering and Computer Science, The Australian National University (ANU), Canberra, ACT 0200, Australia.

I. B. K. Manuaba is with the Commonwealth Science Industrial and Research Organization in ICT Center Canberra and School of Computer Science, College of Engineering and Computer Science, The Australian National University, Canberra, ACT 0200, Australia (phone: +61-2 62167105; fax: +61-2 62167111; e-mail: bagus.manuaba@csiro.au or bagus.manuaba@anu.edu.au).

K. Taylor, is with the Commonwealth Science Industrial and Research Organization in ICT Center Canberra. (e-mail: ken.taylor@csiro.au).

T. Gedeon is with School of Computer Science, College of Engineering and Computer Science, The Australian National University, Canberra, ACT 0200, Australia (e-mail: tom.gedeon@anu.edu.au or tom@cs.anu.edu.au).

Many mixed reality applications have been developed to enable a person to interact with the real world in ways not previously possible, for example in medical visualization [3], mobile phone gaming applications [4] and virtual studios for TV production [5]. Currently most mixed reality applications use the term 'augmented reality', which augments the video with synthetic electronic data. It is difficult to find an example of a mixed reality application which does the opposite, that is to enhance the virtual environment with data from the real world ('augmented virtuality'), especially for teleoperation in mining.

In the context of teleoperation, mixed reality can be used as an interface that mixes the different pathways of visualization, direct visualization through video and synthetic visualization derived from a dynamic software model of the state of the world [6]. One of the requirements for successful teleoperation is situational awareness through visualization. The mixed reality environment creates opportunities for an interface based on information from the virtual model, a camera view from a remote location, and the manipulation of objects within the interface. This includes real objects that have been visualized directly or synthetically, or virtual objects that have been added explicitly for the purpose of remote interaction.

Most current teleoperation systems especially those used in the mining industry contain a number of custom-built user interfaces: As Duff et al. [6] discussed, "typically one for each mining process that needs to be monitored, an alternative to reduce the cognitive load of switching from one interface to another is to present the operator with a single interface". This interface should be interactive and reconfigurable. This can be achieved using platforms built for creating virtual environment computer games.

This paper reports on an experiment to assess the effectiveness of two gaming environments (*Second Life* and *Simmersion's Mycosm*) as a mixed reality interface for a teleoperation system that uses supervisory control to avoid issues associated with latency or low communications bandwidth[7, 8]. Supervisory control means an operator defines a task, which is perhaps simulated locally then assigns the task to the system to undertake autonomously[9]. In these experiments the task assigned was a low level task of move to a position and report the result.

## II. A Teleoperation Interface Using a Mixed Reality Framework

The basic environment of our interface is a virtual reality environment with an augmented virtuality concept added. The information from the real world was used and combined into a virtual world to give information on a remote location. In order to describe the model system of our experiment design, we adopted the LiSA (Localization and Semantics of Assistance) model. This is a common model for teleoperation systems. It defines a relationship between the operator, the interface, the network systems, the manipulator and the environment. This model also can be used to describe a virtual reality system which implements synthetic worlds[10].

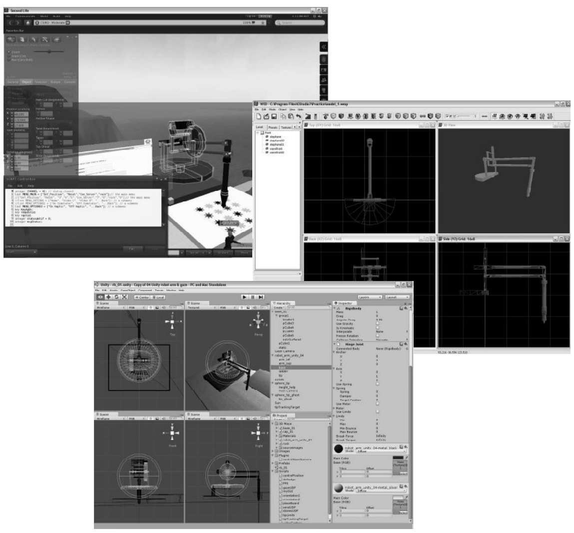

Fig. 1. Left to right: Second Life Viewer, Unity Editor, GED - 3D Game Editor

A virtual game environment can be used as an alternative to a custom built application for a teleoperation interface [11]. There are a number of virtual game environments which have similar characteristics to teleoperation interfaces, such as *Second Life, Unity* and *Game Studio* (shown in Fig. 1). The platforms can meet all of the prerequisites for teleoperation interfaces such as 3D virtual models for visualization, communication into and out of the system and a sophisticated programming environment.

As a model in our experiment, we used a robot arm as a remote device (shown in Fig. 2). Then, we built the 3D model robot arm inside the virtual world as a replica and used the pose of the real robot arm to define its pose in the virtual world. We also provided video streaming from a camera in the real world to show elements of the scene not represented in the model and allow operators to check the model accuracy. The video streaming works by replacing the texture on a surface inside the virtual world. We overlaid the video with a pointer showing the measured position of the robot tip determined by calculating the inverse kinematics from the measured joint angles and projecting the three dimensional tip positions onto the plane of the video.

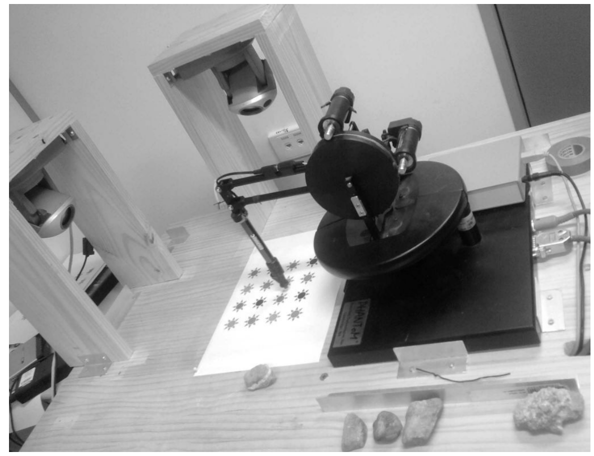

Fig. 2. Real Robot Arm and the PTZ camera.

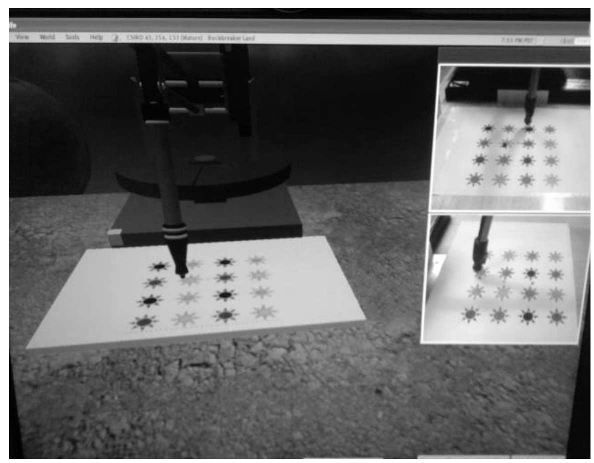

Fig. 3. 3D model Robot Arm in Second Life (SL).

In this experiment we built and compared two different interfaces. The first interface was built by using an online virtual world called *Second Life (SL)* from *Linden Lab*. As shown in Fig. 3, the model robot arm was built in an arena marked with various colored stars which acted as target positions. The interaction between the operator and the model was via an avatar. The operator controls the avatar to move the pointer by clicking and dragging the model pointer. Two video views were also provided from the remote location.

Fig. 4 shows a similar model built using a different platform called *Simmersion (Sm)*. The environment was coded in C++ and made use of the *Mycosm* Library from *Simmersion Holdings Pty Limited* to build the virtual environment and interface. The operator interacted with the

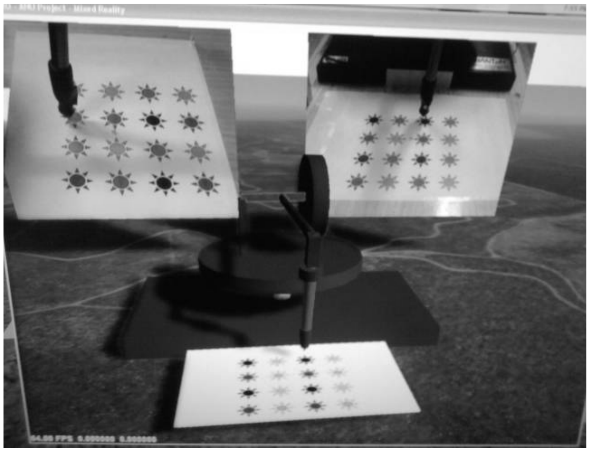
Fig. 4. 3D model Robot Arm in Simmersion (Sm).


Fig. 5 Real Robot Arm Rock breaker

model and once the model was placed in a location, the robot moved to match the model. Communication between the operator machine and robot arm's server was direct, in contrast to the communication between the *SL* Viewer and the server robot arm which went through the *SL* server.

The pointer moves in accordance with the tip position for both interfaces. In the *Sm* interface, the pointer moves as the model moves. However, in *SL* the script that moves the pointer can only run on completion of the user action, so the pointer is updated after the new robot position is defined. In the *SL* interface, the operator can also move the robot arm by clicking on the video. The robot will move in a plane perpendicular to the video camera lens axis and the two cameras are perpendicular to allow movements to be commanded in any direction.

The camera from the virtual world enables the operator to see from any direction. However, the view of the real scene from both cameras is fixed as would usually be the case in a teleoperation application which sometimes allows a camera to pan and tilt but for ease of implementation rarely to be repositioned.

## III. User Study

We based our experiment on a real implementation for *teleoperating* a large rock breaker (Fig. 5) in iron ore mining reported by Duff et al. [6]. The rock breaker was over 1000 km from the operator and was provided with a mixed reality interface. In their trial, the operator relied mainly on the camera views perhaps because the video seemed more real to the operator than the model.

Fig. 6 illustrates an overview of our experiment's architecture. This setting is designed to emulate as closely as possible the teleoperation setting of the real machine which being in a high volume production environment is unavailable for testing. In this experiment, the operator will use a standard input device such as keyboard and mouse as inputs to control the 3D model and the virtual camera in the interface.

### *A. Participants*

A total of 19 volunteers (12 male, 7 female) participated in the experiment all of which are students in various fields at the university. They range from 18 to 49 years of age (Mean 22.9, SD = 7.8). All 19 subjects were regular computer users with no previous experience in this prototype system. Nine of them played computer games "often" (more than one hour per day), another six played computer games "occasionally" and the remaining four never played computer games. Moreover, all subjects did not have color blindness.

### *B. Apparatus and Implementation*

We used the *Second life viewer* version 1.23 for the first interface and another application client was built from the *mycosm* library for the second. These interfaces were run on a desktop PC with the following specifications: NVIDIA *Quadro* FX 1700 for VGA and 2814 MB in memory RAM.

For the main input to the interfaces, we used a standard keyboard and a Dell optical mouse. Based on a gaming control standard system, we set the four key arrows or 'A','W','S','D' as right, up, down, and left respectively; and keys 'E','C' or 'Page up', 'Page down' as zoom in and zoom out to control the virtual camera from an avatar's viewpoint. Holding the 'Alt' key together with 'left clicking' the mouse will allow the virtual camera to be repositioned. To control the robot arm, 'left clicking' the mouse will allow the robot tip to be moved forward, backward, right and left; depressing the 'Ctrl' key and 'left clicking' the mouse will allow the mouse to control the height of the tip.

The display used a standard 19" monitor with a resolution of 1280x1024 pixels to run the interfaces. In designing the remote location, the small robot arm was used

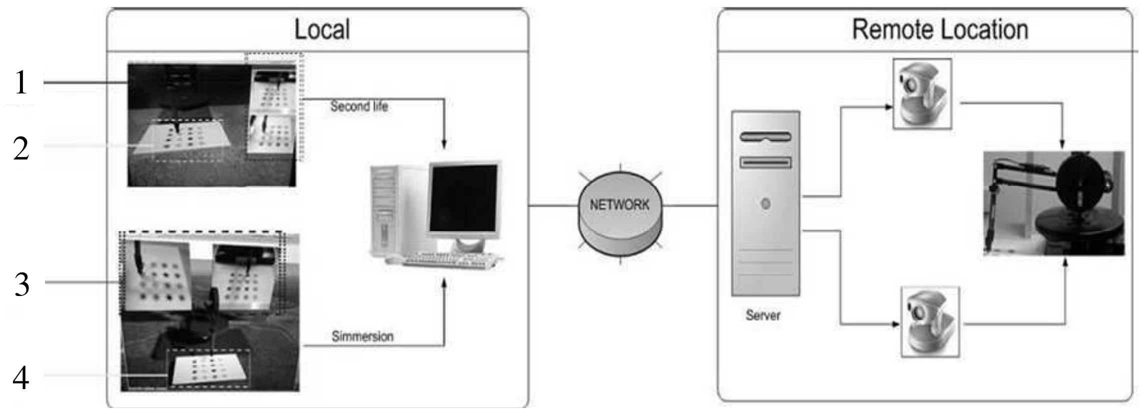


Fig 6. An overview of the experimental architecture, (1 & 3) Video streaming from the camera, (2 & 4) 3D model white board.

as a rock breaker model, observed with two Canon cameras type VB-C50iR. One camera was placed 30 cm in front of the centre of the robot's workspace and 25 cm above the work surface. The other was placed 10 cm to the right side of the centre of the robot's workspace and 25 cm above the work surface. Both cameras pointed to the centre of the scene. The robot was controlled by a networked server.

A white board with 16 pictures of stars in four different colors (red, yellow, black, green) was used as an arena (shown in Fig 2), and each star of the same color was placed 10 cm apart. The function of the pictures is to serve as a target for the robot arm.

On the client site, the interfaces were built with a 3D model of the robot arm, two video displays from the two cameras, and the board model which has the same picture and color as the real board. Items 1 and 3 in Fig. 6 show the video view in the interfaces, while items 2 and 4 in Fig. 6 show the 3D model of the board in the interfaces. (Fig. 3 shows the *SL* interface and Fig. 4 shows the *Sm* interface).

For the experiment, the user gives instructions through each interface to specify a target position for the 3D model with supervisory control concept applied. Verna's [10] LiSA model concept was adopted to provide assistance by combining video overlay inside the virtual game environment.

### C. *Experimental Design and Procedure*

The experiment required the operator to undertake the same two tasks with two different interfaces and subjects alternated which interface they used first. The first task (task 1) was to move the arm to a target position of one color in a clockwise then counter clockwise direction. Each target position must be reached within a maximum time of 30 seconds. The total number of targets for each direction was four. In this task all of the information required to complete the task was available in the model and the video view. The second task (task 2) was to push some rocks continuously to four different target positions. Each target position must be reached within two minutes. In this task the rock position was not modeled and could be determined only from video view.

In the beginning of the experiment, subjects were given a short verbal introduction including brief description about the interfaces; instruction on how to use the interfaces, and the tasks (around 5 minutes). All subjects were required to confirm their understanding of the use of the two interfaces and tasks. Prior to the experiment, pre-training for each interface was provided for approximately 5 minutes. For each interface, subjects were asked to perform the two tasks. As an objective measurement, we noted the times taken to reach each target position. While for the subjective assessment, a questionnaire with Likert Scale (ranging from 1 – 5) was used to determine the subjective time needed to become familiar with the interface, the interface user friendliness and the subjects' perception of the interface performance.

## IV. Result of Experiment

The T-test method with equal variances was applied to compare the average time taken to complete tasks from both interfaces. We define a null hypothesis: the time to complete task in the *SL* interface is not significantly different from the *Sm* interface for each task. In other words, the difference between the means for the two groups is zero.

From table 1, it can be seen that at the 95% confidence level, we did not find any significant difference in task completion time between Second Life and Simmersion ($p>0.05$). However, it could also mean that any difference that may exist is small and would require a more powerful test (i.e. more subjects) to detect at a reasonable level of significance.

In contrast table 2 shows that task completion times with Simmersion were significantly faster than with Second Life (*Sm* : *SL* = 51.16 : 72.34 seconds, $p < 0.05$).

TABLE 1. AVERAGE TIME MEASUREMENT FOR EACH TARGET POSITION IN TASK 1

| Time (seconds) | Task 1 |
|---|---|
| Second Life (SL) | Mean = 18.00<br>SD = 10.87 |
| Simmersion (Sm) | Mean = 14.61<br>SD = 6.45 |
| | t = 1.17<br>p = 0.25 |

TABLE 2. AVERAGE TIME MEASUREMENT FOR EACH TARGET POSITION IN TASK 2

| Time (seconds) | Task 2 |
|---|---|
| Second Life (SL) | Mean = 72.34<br>SD = 25.73 |
| Simmersion (Sm) | Mean = 51.16<br>SD = 26.71 |
| | t = 2.49<br>p = 0.02 |

Fig. 7 shows the result of the subjective measurement for several factors such as difficulty of learning, user friendliness, and interface performance. The bar chart in Fig. 7 shows that the *Sm* interface was regarded as a somewhat quicker to learn. For ease of use, neither interface was obviously preferred over the other. However, interface performance was subjectively assessed as better for the *Sm* interface. This led to a few more people preferring the *Sm* interface over the *SL* interface as shown in Fig. 8.

In spoken and written comments several subjects mentioned difficulty relating information from virtual camera to that from the real camera when the virtual viewpoint was from a very different direction to the actual camera.

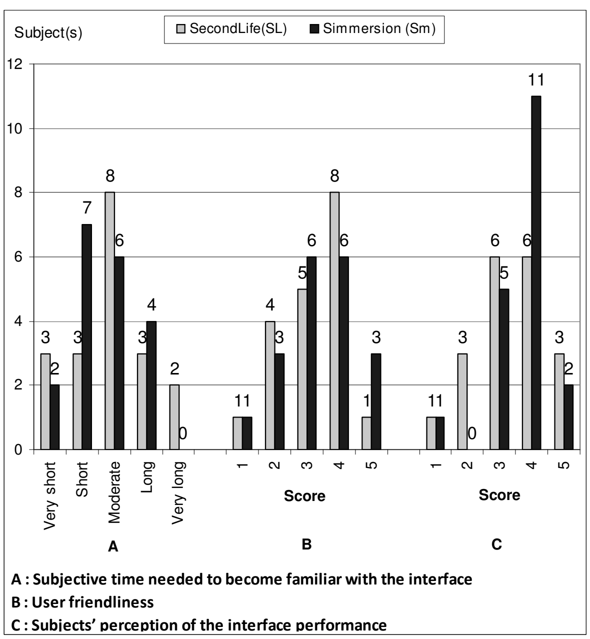

Fig 7 Subjects' assessment of time needed to become familiar with the interface, user friendliness and interface performance.

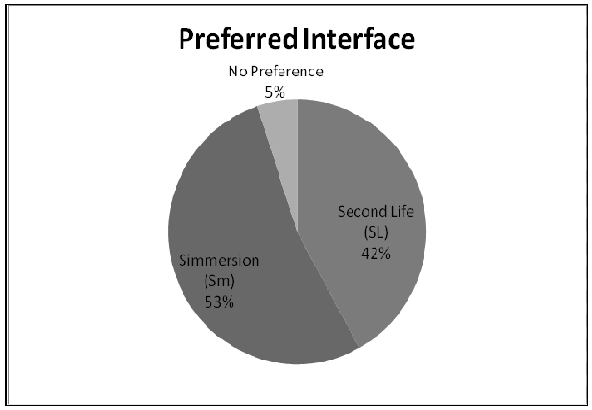

Fig 8. Preferred interface

## V. DISCUSSION

Task 1 was sufficiently well modeled in the virtual world that all of the information needed to complete the task was available without referring to the video. The task could be undertaken paying attention only to the video or looking only at the model.

Observation suggested only occasional reference was made to the video which is in contrast to the real rock breaker interface[6] where the operators attention was predominately directed to the video and the model was mostly not used. Another difference was that in *SL* the operator was represented by an avatar and in *Sm* the operator was not represented as being part of the scene. Our experiment was not able to detect a significant difference in the task performance times suggesting an avatar or any other interface differences did not have much impact on ease of use. An avatar is not a hindrance to performance in this situation but is likely to be an advantage for interactions when two operators must work together on a task. Other experiments have shown that using a three dimensional input device to manipulate a three dimensional model improved task performance but the improvements will apply equally to either interface tested or an interface using purely video feedback [12].

In contrast task 2 where the rocks were not modeled involved a situation where all of the information needed to complete the task was not available in the virtual environment. This is necessarily the situation in most teleoperation scenarios. Building models that contain all of the necessary information requires an excellent understanding of the task to choose what information needs to be collected from the large number of things that could be measured and methods for sensing the required information. This level of knowledge is close to that required to fully automate the task which if it is feasible obviates the need for teleoperation. In task 2, performance was better using the *Sm* interface at the 95% confidence level. The main difference was that in the *Sm* interface the *LiSA* model approach of specifying location was not possible by manipulating the video overlay but in *SL* it was. However in *Sm* manipulations of the model during movement planning were immediately reflected in the video overlay and in *SL* the video overlay was updated after the new location was specified. This was particularly useful when the viewpoint of the virtual camera was from a different direction to the actual camera as subjects found it difficult to understand how object movements would appear from a different viewing direction.

Several subjects commented that when specifying a location using the video overlay they found it difficult to place the robot where they intended in relation to the rock. The converse was not true. When an operator manipulated a 3D model in a virtual world while observing the video overlay to determine the relationship with an object not modeled they were able to place the robot where they intended. Given the higher productivity in task 2 it was surprising that when asked which interface subjects preferred only 11% more people preferred the *Sm* interface, which might be related to the relatively similar assessments both interfaces received for user friendliness, similar time needed to become familiar.

## VI. CONCLUSION

In Summary, both mixed reality environments tested were suitable for teleoperation interfaces where all information to complete a task could be modeled in the virtual world and the effects of avatars, third party servers or other interface differences were not important. Manipulating a model with video overlays to show the effect was a good method of combining video with virtual environments to present information for a teleoperation interface. For the *LiSA* model approach, manipulating a model in a virtual environment and seeing the effect on a video overlay was easier to understand and more effective than specifying a location in a video overlay directly. The *Sm* environment allowed mouse movements to trigger logic that updated the video overlay as objects are dragged where the *SL* interface only generated events when dragging an object was completed and this was an important limitation. Therefore gaming environments to be used for teleoperation should allow logic to be applied during object manipulation.

## REFERENCES


[1] A. Monferrer, and D. Bonyuet, "Cooperative Robot Teleoperation through Virtual Reality Interface," in The Sixth International Conference on Information Visualisation (IV'02), 2002.

[2] P. Milgram, H. Takemura, A. Utsumi *et al.*, "Augmented Reality: A Class of Display on The Reality-Virtuality Continuum" *Telemanipulator and Telepresence Technologies,* vol. SPIE Vol. 2351, 1994.

[3] M. Bajura, H. Fuchs, and R. Ohbuchi, "Merging Virtual Objects with the Real World: Seeing Ultrasound Imagery within the Patient," in SIGGRAPH '92 Chicago, 1992.

[4] O. Rashid, W. Bamford, P. Coulton *et al.*, "PAC-LAN: Mixed-Reality Gaming with RFID-Enable Mobile Phone.," *ACM Computers in Entertainment,* vol. 4, no. 4, 2006.

[5] G. Thomas, "Mixed Reality Techniques for TV and their Application for On-Set and Pre-Visualization in Film Production," pp. 6, 2009.

[6] E. Duff, C. Caris, A. Bonchis *et al.*, "The development of a Telerobotic Rock Breaker," in 7th International Conference on Field and Service Robots (FSR 2009), Cambridge, Massachusetts, USA, 2009.

[7] T. Fong, N. Cabrol, C. Thorpe *et al.*, "A Personal User Interface for Collaborative Human-Robot Exploration," in International Symposioum on Artificial Intelegence, Robotics, and Automtion in Space, Montreal, Canada, 2001.

[8] C. Domingues, S. Outmane, F. Davesne *et al.*, "A Distributed Software Architecture for Collaborative Teleoperation based on a VR Platform and Web Application Interoperability," 2008.

[9] T. B. Sheridan, *Telerobotics, Automation, and Human Supervisory Control*, London, England: The MIT Press Cambridge, Massachusetts, 1992.

[10] D. Verna, "Virtual Reality and Tele-operation: a Common Framework," 1999.

[11] I. B. K. Manuaba, K. Taylor, and T. Gedeon, "Exploration of Second Life as a Virtual Game Environment for Teleoperation," in APCHI Bali, 2010.

[12] F. Nebout, K. Taylor, and S. Mendis, "Gesture Based Interface for Robot Control," in APCHI2010, Sanur, Indonesia, 2010, pp. 680-685.