ENHANCING VIRTUAL ENVIRONMENT-BASED SURGICAL TEAMWORK TRAINING WITH NON-VERBAL COMMUNICATION

Stefan Marks, Burkhard Wünsche
Department of Computer Science, The University of Auckland, Auckland, New Zealand

John Windsor
Department of Surgery, The University of Auckland, Auckland, New Zealand

Keywords: Virtual environment, Non-verbal communication, Image processing, Training.

Abstract: Virtual reality simulations for training surgical skills are increasingly used in medical education and have been shown to improve patient outcome. While advances in hardware and simulation techniques have resulted in many commercial applications for training technical skills, most of these simulators are extremely expensive and do not consider non-technical skills like teamwork and communication. This is a major drawback since recent research suggests that a large percentage of mistakes in clinical settings are due to communication problems. In addition, training teamwork can also improve the efficiency of a surgical team and as such reduce costs and workload.

We present an inexpensive camera-based system for capturing aspects of non-verbal communication of users participating in virtual environment-based teamwork simulations. This data can be used for the enhancement of virtual-environment-based simulations to increase the realism and effectiveness of team communication.

1 INTRODUCTION

In 1999, the Committee on Quality of Health Care in America released a report estimating that each year, 98,000 people die because of medical errors occurring in hospitals (Kohn et al., 1999). This report caused massive changes in the medical education system, including the increased use of medical simulation for training and assessment. As a result, the market now offers a variety of simulators, ranging from inexpensive bench models made of rubber for training suturing and other basic technical skills to expensive mannequins that can be used for the training of whole emergency response teams.

With realism improving through increasing computational and graphics power, computer based tools have also secured their position among medical simulators (Scott et al., 2008). Numerous research projects have contributed to mathematical models for the realistic simulation and manipulation of soft tissue. Based on these models and increasingly realistic rendering capabilities of modern graphics cards, commercially available products offer training opportunities in areas like endoscopic procedures (Undre and Darzi, 2007), obstetrics, cardiology or other related fields.

But regardless of the technical features, these simulators only serve the purpose of training a single person at a time. Surgery and other medical procedures are always performed as a team and thus, among the technical skills, require non-technical skills like communication, teamwork, leadership, and decision making (Yule et al., 2006). Research indicates that failure in communication occurs in 30% of team exchanges and that one third to 70% of these failures result in situations putting the patient's life at risk. (Lingard et al., 2004; Guise and Segel, 2008). To increase patient safety, training and improvement of communication among team members has to be an important aspect of clinical simulation.

2 RELATED WORK

Computer based teamwork simulation has been used by the military and the aviation industry for years. Only recently, this kind of simulation has addressed the area of healthcare and medicine.

3DiTeams (Taekman et al., 2007) simulates a military operating theatre, including all necessary de-

Marks S., WÃijnsche B. and Windsor J.
ENHANCING VIRTUAL ENVIRONMENT-BASED SURGICAL TEAMWORK TRAINING WITH NON-VERBAL COMMUNICATION.
DOI: 10.5220/0001793703610366
In *Proceedings of the Fourth International Conference on Computer Graphics Theory and Applications (VISIGRAPP 2009)*, page
ISBN: 978-989-8111-67-8

vices, medical instruments, and the patient. In the virtual environment (VE), team members are represented by avatars and can interact with each other, the devices, and the patient. Each user operates at an individual client computer that is connected to a central server by a network. This server receives all actions and movements of the users from the clients, runs the physical simulation of objects and medical simulation of the patient's physiology, and synchronises the updated state of every user and object on the clients.

The simulation uses a game engine as underlying framework. Such an engine provides a well-tested and stable base capable of realistic graphics, animation, sound, physics and networking support (Marks et al., 2007). The principle of exchanging the content of a computer game by content with educational or training purpose is referred to as "Serious Gaming" (Serious Games Initiative, 2007).

With the focus on teamwork and communication aspects, it is important to examine the amount of communication channels that the simulation provides for the users (see table 1 in section 3.1). Modern game engines support verbal communication via microphones and headsets or speakers or predefined textual commands that are accessible by pressing certain keys.

Despite the advances in technology, the support for other communication channels is still limited. This renders communication and interaction within VEs inefficient and unnatural (Vertegaal et al., 2001; Garau et al., 2001). Without gaze direction and head orientation, for example, communication suffers from a decrease of 50% in deictic references to persons, like "you" or "him/her" (Vertegaal et al., 2000). The lack of non-verbal communication channels has to be compensated by other channels, for example, by replacing deictic references to objects and persons by explicitly saying their names and positions (see figure 1).

Figure 1: Without gaze direction and head orientation, communication has to use unnatural verbal reference to objects or people.

Another method of compensation is described in (Manninen, 2001). Without the implementation of gestures in the game "Counter Strike", users cannot refer to a location or a direction by, for example, pointing a finger into a certain direction. In this case, the lack of gesture is compensated by turning the avatar into the desired direction and moving then forwards and backwards in rapid succession. Asked for improvements of the game, the introduction of gesture was an important aspect for the users.

By introducing, for example, head orientation and gaze direction, users can simply look at objects or other users instead of referring to them by voice (see figure 2).

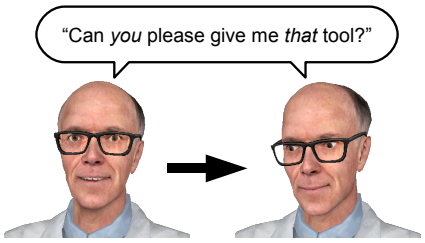

Figure 2: With gaze direction and head orientation, the verbal reference to objects and people is not necessary any more.

Other capabilities of modern game engines or VEs allow for even more non-verbal communication channels that can be used to increase the bandwidth. Detailed animation systems in the avatar's face models enable the rendering of finely controlled facial expressions like joy, anger, fear (see Figure 3). Partial animations that can be applied independently could be used for the control of hand or body gestures like pointing, turning, or shrugging shoulders.

Figure 3: Changing an avatar's facial expression to represent emotions like joy, anger, fear.

Instead of using explicit user interface controls for changing facial expression, triggering gestures, or controlling gaze direction (Slater et al., 2000), we propose the use of a camera to extract the necessary data in real-time. This has several advantages. Manual control of the aspects of non-verbal communication is only possible if the user is aware of them, which is not necessarily the case. The camera requires no control from the user and can capture conscious and unconscious elements. In addition, the user can completely concentrate on the simulation content instead of getting distracted by the additional controls. A second advantage is the temporal immediacy of the captured

data. Momentarily raised eyebrows during an emphasis in a spoken sentence can be perceived by all others users at the same time. If the optical clue would follow the verbal clue with a delay, for example, when using manual control, it would degrade or even counteract the purpose of the gesture.

We present a method for the enhancement of VE based teamwork simulations with aspects of non-verbal communication that can be captured simply by using an inexpensive camera.

3 METHODOLOGY

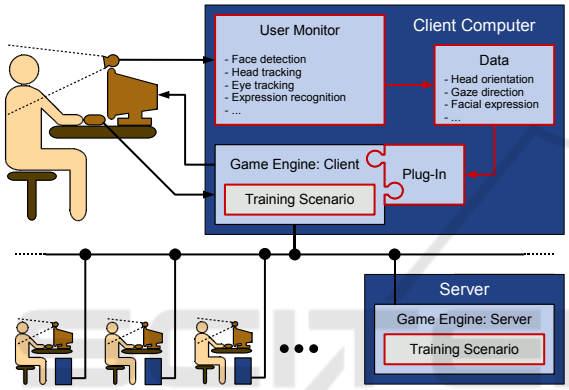

Figure 4: The functional blocks of our framework. Red borders indicate the parts that have been developed by us.

A schematic design of our system is depicted in figure 4. The user participating in the simulation is captured by a webcam, mounted close to the monitor. A user monitor application detects the user's face in the video and calculates parameters like head orientation, gaze direction, and facial expression. Via an extension, called plug-in, the application provides this data to the simulation client, in this case a game engine. The client sends this information to the simulation server together with other data, for example, mouse movement, or pressed keys. The server receives the data from all clients, applies it to the simulation and in turn synchronises all clients with the updated information, including the aspects of non-verbal communication. The clients receive these updates and display it in form of changed positions of objects, a changed patient state, and changed gaze directions and facial expressions of the avatars.

3.1 DataModel

Based on the Rich Interaction Model (Manninen, 2004), we have derived a list of aspects of communication in real and virtual environments and constructed a data model defining parameters necessary to represent these aspects. The model also provides the data in different levels of abstraction to allow for a wider range of simulation engines to be connected to (see section 4).

Table 1 lists the components of this model, their occurrence in reality, and the technical feasibility in VEs, given the current state of computer technology. Not all aspects of the real world can be implemented in the virtual world, for example, olfactics. Also, not all aspects are equally important and we have to weigh the benefit of adding an aspect against the technological challenge in acquiring the data.

Table 1: Aspects of communication and their appearance in reality ("+": used naturally, "-": could be used, but unnatural) and technical feasibility in virtual environments ("+": available using standard equipment, "-": requires special hardware and software).

Component	Reality	VE
Occulesics		
- gaze direction, duration, focus	+	+
Language		
- text based chat	-	+
- sound effects	-	+
- speech	+	+
- paralanguage (tone, pitch, ...)	+	+
Facial expression		
- Facial Action Coding System	+	+
- emotion (anger, joy, ...)	+	+
Spatial Behaviour		
- orientation, proximity, ...	+	+
Kinesics		
- head movement, gestures, ...	+	+
Physical appearance		
- skin, hair, clothes, ...	+	+
Physical contact/Haptics	+	-
Olfactics (scent, odour)	+	-
Environment		
- artefacts (use, exchange, ...)	+	+

Depending on the choice of the simulation platform we are connecting our user monitor application to, several aspects of the data model are already implemented. A modern game engine usually enables

verbal communication by support of headsets and microphones, spatial behaviour by the movement and orientation of avatars, physical appearance by a selection of avatars that can be adjusted to the user's needs, and environment by physically simulated objects that can be grabbed or moved. Data input for these aspects comes from the keyboard or mouse movement, and does not have to be provided by our user monitor application.

Our framework adds occulesics, kinesics, and facial expression by defining an interface between the application and the simulation that exchanges data about the gaze direction, the head orientation and the facial expression of the user.

By increasing the viewing angle of the camera or placing it further away from the user, an additional module in the user monitor application could capture not only the face but the upper body movement and hand gestures. This could add support for gestures like shrugging of shoulders or pointing to the simulation and increase the communication bandwidth even further.

3.2 Image Acquisition and Processing

Our system utilises an inexpensive camera, for example a webcam, to capture the user's face. The video stream is handled by the user monitor application that allows modular processing of each frame. The modularity enables us to plug in a variety of pre-processing filters to measure their influence on the performance of different feature tracking algorithms.

In a first step, we are applying a colour space conversion from RGB to normalised RGB, as this enables us to use simpler models for the distinction between skin colour and non-skin colour pixels for face detection (Terrillon et al., 1999). Colour changes caused by different lighting, are compensated by an improved Grey World algorithm, described in (Chen and Grecos, 2005).

For the detection of features in the video stream, we implemented a fast normalised cross correlation module (Briechle and Hanebeck, 2001). This algorithm uses the concept of the integral image together with the decomposition of the feature template into a sum of k basis functions to achieve a computational complexity of $O(M_x \cdot M_y \cdot k)$. Further reduction of the computation time is achieved by reducing the Region of Interest (ROI) for each template (see the coloured regions in figure 5).

Using the input of the cross correlation module, we are able to calculate a rough estimate of the spatial orientation of the head. This estimate can be refined by applying advanced mathematical models like Kalman filtering, applying constraints that take into account the symmetry of a face, and additionally tracked features.

If the head orientation is known, the captured image of the face can be normalised and gaze direction and facial expression features can be extracted (Hammal et al., 2005; Whitehill and Omlin, 2006).

The game engine we connected the user monitor application to (see section 3.3) supports facial animation in high detail. Based on the FACS (Facial Action Coding System) (Ekman et al., 2002), a range of controllers enables fine grained control of the face of the avatar. But not all simulation engines are able to support facial animation in such detail. In that case, it might be necessary to interpret the acquired data and to create an abstract representation of the facial expression, like, for example, "happy", "angry". Our data model is designed for these different levels of abstraction.

3.3 Interfacing with the Simulator Client

When the user monitor application has processed a camera frame and calculated a set of data, it passes this set on to the simulation client, using a plug-in. The plug-in is an addition to the simulation engine that translates the data model into a specific representation that considers the graphical and other capabilities of the engine.

We connected the user monitor application to the Source Engine (Valve Corporation, 2004), a modern, versatile, commercial game engine with advanced graphical features, allowing us to make full use of the potential of our data model.

The client receives the data and sends it, together with the input from the mouse and the keyboard, in form of a compressed data packet to the server. Each server entity of a user's avatar gathers the input from its specific client entity. Then the simulation of the movement of the avatars, the interaction with the simulated physical environment and other simulation modules, for example for the patients biological parameters, are executing one simulation time step. Afterwards, networked variables automatically synchronise the state of the simulation on each client. The clients then display the updated state of the VE.

In order to minimise network traffic and to maximise the simulation frame rate, game engines generally optimise data transfer between server and clients by compression, either by reducing the precision of variables or by transferring only changes. In the case of the Source Engine and the majority of other VEs, the data model has to be transferred three times in

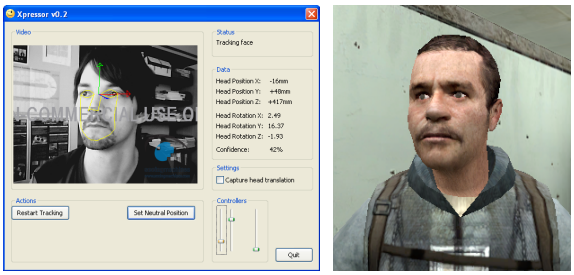

Figure 5: Screenshot of the video capture of the user participating in the VE simulation. The head tilt and eye gaze are transferred onto the avatar representing the user in the simulation.

different levels of compression (user monitor application to client, client to server, server to clients). A high level of complexity would result in having to extend the data exchange mechanisms of the engines. To maximise the re-use of components the VE engine provides, we constructed the data model from non-complex data types. As a result, the necessary changes to the code of the Source Engine were minimal, but nevertheless allow for flexibility of the data model to include future changes or extensions.

4 RESULTS

Figure 5 depicts our user monitor application connected to a simple dummy simulation. In the left image, the position and the orientation of the face is indicated by the yellow overlay and the coordinate axes. This information and additional parameters, like gaze direction or facial expression, are transferred to the simulation. The right screenshot shows how other users participating in the simulation can see the head tilt and gaze direction projected on the avatar.

Our data model is universal and independent of the VE it is connected to. If support for detailed facial expression is provided, for example, if the VE implements FACS or the MPEG-4 standard, the corresponding parameters of the data model can be used with only minor adaptions. This is the case for the majority of modern game engines, for example, the Source Engine (Valve Corporation, 2004) or the Unreal Engine 3 (Epic Games, 2006). If the support is not as fine, another abstraction layer, for example, *interpreted* data model parameters (see sections 3.1 and 3.2) can be used for an alternative way of displaying the emotional state of the user, for example by changing the texture of the avatar's face. This flexibility of our data model enables the connection to a wide range of VEs and game engines.

An example for limited support of facial expressions is Second Life (Linden Research, Inc, 2008). This VE has gained much popularity in the last years and is also increasingly used for teamwork training (Ullberg et al., 2007). The control provided by Second Life over the avatar's facial animation is not as fine grained as for modern game engines and is limited to displaying pre-defined animations. In this case, the interpreted parameters of our data model can be used to display basic emotions in form of short animations.

5 CONCLUSIONS AND FUTURE WORK

We have presented a framework that allows the enhancement of VE based teamwork simulations by non-verbal communication. The necessary data is captured in real-time by an inexpensive webcam. Our framework is flexible, extendible and independent of the used simulation engine.

We have received positive feedback from medical professionals and developers of teamwork simulations in Second Life about the use and the potential of our application. and will perform a more detailed user study to measure the increase of communication bandwidth, to verify the ease of use of the camera based data acquisition, and to improve and extend our data model.

In cooperation with researchers and educators from the medical school, we are going to design surgical training scenarios to be used with our application. Furthermore, our user monitor application and the data model is suitable for teamwork training applications beyond the field of medicine and surgery. For this reason, we are also in cooperation with developers for emergency response teamwork training in Second Life, giving us the opportunity to collect valuable information with their simulation scenarios.

The stability of the facial recognition and feature tracking algorithms will be subject to further investigation. Several facial recognition algorithms require an extensive training phase that we would like to eliminate or at least hide as much as possible from the end user. Also, we will examine how to overcome difficulties in the image processing caused by non-ideal lighting, users wearing glasses, or other circumstances.

Another goal is the integration of our application with the Unreal Engine that is increasingly used for simulations (Virtual Heroes, 2008). The recent version 3 of this engine is capable of displaying very realistic facial animation and human skin, allowing us to fully apply the range of parameters of our data model.

REFERENCES

Briechle, K. and Hanebeck, U. D. (2001). Template matching using fast normalized cross correlation. *Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE)*, 4387(95):95–102.

Chen, L. and Grecos, C. (2005). A fast skin region detector for colour images. *IEE Conference Publications*, 2005(CP509):195–201.

Ekman, P., Friesen, W. V., and Hager, J. C. (2002). *The Facial Action Coding System – Second Edition*. Weidenfeld & Nicolson.

Epic Games. Unreal Engine 3 [online]. (2006) [cited 17.08.2007]. Available from: http://www.unrealtechnology.com/html/technology/ue30.shtml.

Garau, M., Slater, M., Bee, S., and Sasse, M. A. (2001). The impact of eye gaze on communication using humanoid avatars. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 309–316, New York, NY, USA. ACM.

Guise, J.-M. and Segel, S. (2008). Teamwork in obstetric critical care. *Best Practice & Research Clinical Obstetrics & Gynaecology*, 22(5):937–951.

Hammal, Z., Massot, C., Bedoya, G., and Caplier, A. (2005). Eyes Segmentation Applied to Gaze Direction and Vigilance Estimation. In *Pattern Recognition and Image Analysis*, volume 3687/2005 of *Lecture Notes in Computer Science*, pages 236–246. Springer Berlin, Heidelberg.

Kohn, L. T., Corrigan, J. M., and Donaldson, M. S., editors (1999). *To Err is Human: Building a Safer Health System*. National Academy Press, Washington, DC, USA. Available from: http://www.nap.edu/catalog.php?record_id=9728 [cited 17.04.2007].

Linden Research, Inc. Second Life [online]. (2008) [cited 08.09.2008]. Available from: http://secondlife.com/.

Lingard, L., Espin, S., Whyte, S., Regehr, G., Baker, G. R., Reznick, R., Bohnen, J., Orser, B., Doran, D., and Grober, E. (2004). Communication failures in the operating room: An observational classification of recurrent types and effects. *Quality & Safety in Health Care*, 13(5):330–334.

Manninen, T. (2001). Virtual Team Interactions in Networked Multimedia Games – Case: "Counter-Strike" — Multi-player 3D Action Game. In *Proceedings of PRESENCE2001 Conference*.

Manninen, T. (2004). *Rich Interaction Model for Game and Virtual Environment Design*. PhD thesis, University of Oulu, Finland.

Marks, S., Windsor, J., and Wünsche, B. (2007). Evaluation of Game Engines for Simulated Surgical Training. In *GRAPHITE '07: Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, pages 273–280, New York, NY, USA. ACM.

Scott, D. J., Cendan, J. C., Pugh, C. M., Minter, R. M., Dunnington, G. L., and Kozar, R. A. (2008). The Changing Face of Surgical Education: Simulation as the New Paradigm. *Journal of Surgical Research*, 147(2):189–193.

Serious Games Initiative. Serious Games Initiative [online]. (2007) [cited 17.08.2007]. Available from: http://www.seriousgames.org.

Slater, M., Howell, J., Steed, A., Pertaub, D.-P., and Gaurau, M. (2000). Acting in Virtual Reality. In *Proceedings of the Third International Conference on Collaborative Virtual Environments*, pages 103–110.

Taekman, J., Segall, N., Hobbs, E., and Wright, M. (2007). 3DiTeams – Healthcare Team Training in a Virtual Environment. *Anesthesiology*, 107(A2145):A2145.

Terrillon, J.-C., Fukamachi, H., Akamatsu, S., and Shirazi, M. N. (1999). Comparative Performance of Different Chrominance Spaces for Color Segmentation and Detection of Human Faces in Complex Scene Images. In *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*, pages 54–61.

Ullberg, L., Monahan, C., and Harvey, K. (2007). The New Face of emergency preparedness training: using Second Life to save first lives. In Livingstone, D. and Kemp, J., editors, *Second Life Education Workshop 2007*, pages 96–99. Available from: http://www.simteach.com/slccedu07proceedings.pdf.

Undre, S. and Darzi, A. (2007). Laparoscopy Simulators. *Journal of Endourology*, 21(3):274–279.

Valve Corporation. Valve Source Engine Features [online]. (2004) [cited 17.08.2007]. Available from: http://www.valvesoftware.com/sourcelicense/enginefeatures.htm.

Vertegaal, R., Slagter, R., van der Veer, G., and Nijholt, A. (2001). Eye gaze patterns in conversations: there is more to conversational agents than meets the eyes. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 301–308, New York, NY, USA. ACM.

Vertegaal, R., van der Veer, G., and Vons, H. (2000). Effects of Gaze on Multiparty Mediated Communication. In *Graphics Interface*, pages 95–102.

Virtual Heroes. Virtual Heroes Inc – Serious Games and Andvanced Learning Systems [online]. (2008) [cited 04.09.2008]. Available from: http://www.virtualheroes.com

Whitehill, J. and Omlin, C. W. (2006). Haar Features for FACS AU Recognition. *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pages 97–101.

Yule, S., Flin, R., Paterson-Brown, S., and Maran, N. (2006). Non-technical skills for surgeons in the operating room: A review of the literature. *Surgery*, 139(2):140–149.