Interaction Methods for Virtual Reality Applications

Leonhard Vogelmeier, Harald Neujahr and Dr. Peter Sandl
EADS Deutschland GmbH, 81663 Munich
GERMANY

leonhard.vogelmeier@eads.com / harald.neujahr@eads.com / peter.sandl@eads.com

ABSTRACT

Even after several years of development, human machine interfaces applied in Virtual Reality (VR) environments are in many cases not very well adapted to the user and the task to be fulfilled, which often affects the success of VR applications. This observation is made by plenty of experts and it coincides with the experience we have gathered within the past years.

On this basis we started to adapt and advance existing HMIs, aiming to improve the handling and to meet the special requirements of dedicated applications. Our first VR application was the cockpit development using virtual prototypes. For this exercise it is essential to ensure, that the deviation of the real human body from its virtual representation is within a defined margin. Another important feature for cockpit development is the provision of haptic feedback. Both requirements could not be satisfied using commercially available tools, so we needed to develop our own methods.

In the beginning we concentrated our activities on a precise representation of the human body. For this purpose we designed easy-to-use calibration methods for the measurement of the tracking sensor positions at the human body. In addition we developed a new kinematic model, which was able to compensate for inaccuracies, which arise from differences between the virtual and the real skeleton. In order to provide haptic feedback we built a flexible Mixed Mock-Up system, parts of which can be adjusted by the user during the VR session.

Changing our focus of research to maintainability and training applications in recent years, we began to design appropriate interaction concepts and to investigate the implementation of the concepts using several interaction methods.

The paper will provide a survey of our activities and present some interesting results.

1.0 INTRODUCTION

Since the time we are engaged in Virtual Reality (VR) we experienced again and again that the potential of this technology is not taken full advantage of, because the user interfaces are not designed to effectively support the user. In this respect we can confirm the observations made by other VR experts (see 1, 2). And one of the most critical interface aspect from our point of view is the possibility to *interact* with the system under consideration within the virtual environment.

The initial idea of optimising the user interaction was to transfer the interaction that occurs in the real world as accurate as possible into the VR environment. In this case the user, theoretically, would have no problems in interchanging between the two worlds. The problem here is, that the VR technology is not yet mature enough to do this. Some elements of the real world interaction, e.g. haptic feedback, cannot be transferred into the VR environment, others only with insufficient precision. Examples for common problem areas are:

Vogelmeier, L.; Neujahr, H.; Sandl, P. (2006) Interaction Methods for Virtual Reality Applications. In *Virtual Media for Military Applications* (pp. 14-1 – 14-8). Meeting Proceedings RTO-MP-HFM-136, Paper 14. Neuilly-sur-Seine, France: RTO. Available from: http://www.rto.nato.int/abstracts.asp.

- haptic feedback,
- operation of controls requiring fine motor skills (e.g. rotary controls),
- the match of the movement of a real person and its virtual representation, and
- the visualisation quality.

1.1 Differences between VR and Reality

Despite the growing possibilities in addressing the human senses with special hardware devices, it is normally not possible to generate an exact copy of the real world. Therefore fundamental differences exist between the real and the virtual world, and to always keep them in mind is essential in designing a VR application which requires minimum conversion effort from the user. We would like to explain these differences using the simple example of holding a ball in two hands.

In reality a ball is kept hold of with two hands, the position of the hands is determined by the surface of the ball and control of the ball is provided by the haptic and the visual senses[1] of the human. In a virtual environment, where no haptic feedback is provided, the control of the ball must exclusively be accomplished by the visual system. There are several aspects which make it much more difficult to control the ball in the virtual environment than in reality:

- the hands seen by the user are not his own hands but more or less realistic reproductions,
- the ball is also only a reproduction,
- the positional quality of the reproduction of both the ball and the hands is in most cases not perfect, and
- the user is not really holding a ball (in fact he is not holding anything), he has to position his real hands according to the visual movement of the reproduction of his hands. This causes a completely different stimulation of his muscles than with a real ball (no weight and no counter-forces from the ball).

A task which is really simple in reality turns out to be very challenging in a VR environment, and the only way to make the user able to deal with these issues is to train him. The same example can also make clear that such a VR application is absolutely inadequate in order to learn how to hold a ball, because what the user trains in VR (how to hold a virtual ball) is only of very limited use if the task is to hold a real ball.

We can learn from this example, that, where noticeable differences exist between reality and the virtual world which cannot be overcome, it is not worthwhile to put a lot of effort into the design of interaction methods which try to mirror reality as exactly as possible. In these cases other ways of interaction should be taken into consideration. Examples from aircraft design are given below.

2.0 VR IN COCKPIT DEVELOPMENT

The first VR application where we faced problems with the existing interaction devices and methods was cockpit development. The basic requirement was to ensure the correctness and accuracy of the data which was generated in the VR environment. We had to make sure, that the findings from our VR exercises were transferable to the real product. So our initial idea was to carry out VR cockpit assessments as we were used to in real aircraft, which meant that we had to reproduce the real aircraft interaction within our VR environment. As an exact reproduction was and still is not possible, we first looked at the maximum difference between VR and reality which is just acceptable for a user and second we investigated if the data gathered in VR was good enough for aircraft development.

[1] The primary role is played by the haptic sense, which makes sure that the ball is not lost, even if the person is not looking at it.

Very soon we identified two problem areas:

- the visualisation error caused by an improper placement of the virtual camera, which is the users eye into the virtual world, and
- the error arising from an inaccurate hand model and an imprecise finger tracking method.

As a start we concentrated on reducing the error in the positioning of the virtual camera. The relevant data is gained by tracking the head of the user. In our case the tracking was done by an optical system and the reflectors necessary for this task have been mounted on the data helmet used for visualisation. It is obvious that the distance and orientation of the reflectors to the users real eyes depend on the way the user wears the helmet. A series of trials showed that with the standard calibration of the helmet more than 40% of our test persons complained about an unnatural visual perception of the cockpit. Looking for the reasons we made another test series, where we found out, that the way users wore the helmet differed by up to 16°. Even the intraindividual variance was up to 7°. So we decided to develop a calibration process, which is executed after the user has taken on the helmet. With the additional calibration we managed to position the virtual camera much better at the users real eye position, independent from the way the user wears the helmet. After we had introduced the new calibration process, there were no more complaints about the perception. We also learned from this exercise, that it is less important for the user that the virtual environment looks like a photograph of the real environment. Much more important is that everything is at the right place.

Haptics play an important role in cockpit development. To be able to assess haptic aspects we reverted to the Mixed Mockup concept, i.e. we built a piece of hardware that is flexible enough to represent all types of military cockpits and tactical working environments. The challenge for Mixed Mockups is to make sure that the visual representation of objects in VR is within ± 2 mm of their real position in order to provide a realistic feeling to the user. As we also wanted to include a representation of both the thumb and the index finger of the user it was necessary not only to reduce the visualisation error of cockpit elements but also of the hand, including index finger and thumb. With the standard dataglove and a standardised hand model it was not possible to achieve the required precision. So we replaced the dataglove by home-made hand/finger/thumb trackers compatible with our optical tracking system. Instead of the standardised hand model we developed a flexible model, which is able to be adapted to the real geometry of the users hand. In addition we implemented a calibration process, which measures, in three steps, the rough dimensions of the hand and the precise lengths of index finger and thumb as well as the position and orientation of the tracking sensors on hand, index finger and thumb.

The two calibration methods contributed to a high level of user acceptance. Even users with absolutely no VR experience were now able to use the system for cockpit assessments without any problems. Our next activity is to expand these tracking and calibration methods to the whole human body.

3.0 VR FOR MAINTENANCE APPLICATIONS

In 2005 we began to concentrate our VR activities on "design to maintainability" and "maintenance training". For these applications a much more complex interaction with the VR environment is necessary than for cockpit development. The user is not a pilot but a technician, who e.g. has to operate tools to turn screws, has to install and dismount objects in difficult to access positions, sometimes in a cramped environment and with uncomfortable posture. The operation and control of virtual objects in a virtual environment with plenty of geometrical constraints is the basic task of a maintenance technician, who has to evaluate constructional solutions or who practises maintenance procedures using VR.

In user tests we evaluated five ways in order to find out which interaction methods are adequate for a maintainability engineer who designs a maintenance procedure, and which methods could be applied to train a student in a maintenance task. The task for each subject was to mount an LRU[2] inside the avionic compartment of an aircraft. In order to reach the final position the LRU has to be rotated several times around various axes. The visual control of the movement was difficult because the aircraft structure often affected the view to the equipment.

3.1 Method One: Contact Simulation without Haptic Feedback

The first alternative was to use a Data Glove for interaction in combination with a contact simulation (3,4). Visualisation was provided by the data helmet and no haptic feedback was available.

To move the LRU the user grabs it with his virtual hand. A grasp occurs if both index finger and thumb of the virtual hand have contact with the LRU. Once held in the virtual hand the position and orientation of the LRU is controlled by the points where the index finger and the thumb touch the LRU. Any movement of the index finger and/or the thumb therefore changes the position and orientation of the LRU. It was also possible to grab the LRU with two hands, wearing two datagloves. In this case the position and orientation of the LRU is controlled by the points where the palms touch the LRU (see also Fig. 1).

The contact simulation ensures that, if a collision between the LRU and the aircraft structure occurs, a compensation movement is calculated. An example shall clarify this movement: normally, if a virtual pen, which is held at one end, collides with a virtual plane, its movement stops immediately. With contact simulation a compensation movement occurs, which, in our example, causes a rotation of the pen and the pen glides along the virtual surface if the movement is continued.

Another characteristic of the contact simulation is that, if a collision occurs and the real hand continues to move, the positions of the real hand and the virtual hand do no longer correspond. The freedom of the virtual hand is restricted by the virtual structure. But, because the VR-System is not able to apply forces to the real hand, its movement can not be limited and so the real hand can be moved to positions that are not reachable for the virtual hand. This positional difference is visualised by the introduction of a second virtual hand in a wireframe look which represents the position of the real hand. (see Fig. 1).

This represents a breach of the correlation between the visual and the kinaesthetic control of the hand, which in reality automatically occurs. With this design the user gets different feedback about the position and orientation of his hand, which makes the control of the virtual hand very difficult.

So it was not surprising that this option only proved to be working for easy movements like gliding the LRU along a plane. If, in contrast, a complicated path was to follow, it turned out that the user soon got stuck in the aircraft structure and was not able to break the deadlock. It was impossible for him to decide, which movement of his real hand releases the virtual hand holding the LRU from the jam. Even experienced VR users failed to move the LRU to its end position. This method is another example for a case, where the interaction method makes a solvable task unsolvable (see the ball example above).

[2] LRU stand for Line Replaceable Unit. It is a box of electronics, such as a radio or other auxiliary equipment for a complex engineered system like an airplane or ship. LRUs speed up repair, because they can be replaced quickly, restoring the big system to service (5).

Figure 1: Holding a Ball with Two Virtual Hands; the Opaque Hands Show the Positions of the Virtual Hands, the Wireframe Hands Show the Positions of the User's Real Hands. They are, in contrast to the virtual hands, able to penetrate the ball.

3.2 Method Two: LRU Tracking without Contact Simulation

The second option was to skip the contact simulation and to utilise a real size model of the LRU made of polystyrene. It was provided with an additional tracking sensor, so that the movement of the real LRU and its virtual representation matched. The user was wearing two datagloves, they were visualised by two virtual hands. So the user was able to see and also *feel* the LRU. This made control of the LRU easy, even for users with no VR experience. Every test person was able to move the LRU to its end position in a very short period of time. However, it has to be kept in mind that the contact simulation was switched off for that exercise. So it was possible for the test persons to move through the aircraft structure, which is, admittedly, not realistic.

This interaction method is certainly not suitable for finding possible installation/de-installation paths. But we think it is useful to train the maintenance technician in his task to identify and become *roughly* familiar with the mounting path. He learns e.g. at which positions and to what extent the LRU has to be rotated in order to be positioned at the correct location. To give the student an estimation about how narrow the room for movement is inside the aircraft structure, he should be informed about a collision by adequate means, such as visual and/or acoustic feedback. The intention of the collision display is not to train the student to follow the mounting path without any collision, but to make him aware that in reality the installation procedure must be performed in a narrow environment.

3.3 Method Three: Force Feedback Device plus Contact Simulation

The third method applied an active force feedback device (FFD), which comprises a robotic arm, the joints of which were controlled by electric motors (see Fig. 2). The FFD allowed to provide a force onto the user. The range of the robotic arm was comparable to the range of a human arm. A mock-up of the LRU was mounted at the end of the FFD arm. The contact simulation linked to the FFD made sure that, in case of a collision between LRU and aircraft structure, the respective force is transmitted to the user. Thus the user was able to perceive any collision in a very reality like manner. The LRU could be grabbed either with one or with both handes. The FFD proved to be suitable to support the movement of the LRU to its end position, but only if all six degrees of freedom (rotation and translation) had been implemented. The limitation to the three translative degrees of freedom was rather disturbing than helpful.

Problematic with the use of an FFD is the high effort for the preparation of the data models and for the force feedback hardware. Also, the FFD we used was not able to deploy enough force to simulate the real weight of the LRU. It was not possible to train the mounting like in reality, because heavy objects are grabbed and handled different than light objects. The method also only allows to train, just like method two, the rough installation movements. The user can only be trained to follow the right path, but the haptic feedback to the muscles is incorrect. Unlike method two the utilisation of an FFD is adequate to find possible mounting paths.

Figure 2: User Working with the 6DOF Force Feedback Device SAMIRA II (6).

3.4 Method Four: Space Mouse with Contact Simulation

The fourth interaction method was to use a SpaceMouse (see Fig. 3) to control the movement of the LRU in combination with a contact simulation. The visualisation was stereoscopic on a desktop monitor. The SpaceMouse allowed the user to move an object in all six degrees of freedom.

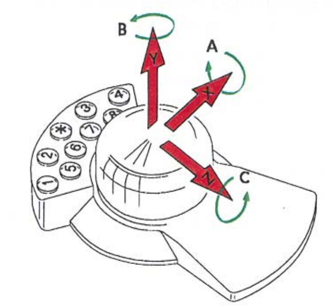

Figure 3: SpaceMouse.

The control in the middle can be pushed in three directions for translatory movements (X, Y, Z) and twisted around three axes for rotations (A, B, C). As the possible displacement of the control is only some millimetres, its displacement controls the speed (not the position) of the movement or the rotation of the controlled object. This way of controlling objects is very different from the way it is done in reality, which makes it unsuitable for training applications but provide a promising option for the maintainability engineer to assess mounting paths.

The usage of the SpaceMouse must be trained before a user can control objects with it. Only highly trained test persons proved to be able to move the LRU to its end position and they usually needed several attempts. Summarising we think that the SpaceMouse is not a real option for VR applications but it may be a useful interaction device for desktop applications like CAD.

3.5 Method Five: Flying Mouse with Contact Simulation

The last method we investigated was to use a FlyingMouse (see figure 4) for the control of the movement of the LRU in combination with a contact simulation. The visualisation was done in a four Side Cave[3]-System. The FlyingMouse has the form of a handle; it contains one or more triggers and a tracking sensor to pick up its position and orientation in space. By pressing a trigger, the virtual LRU is tied to the FlyingMouse and is moved according to the movement of the FlyingMouse. So if the FlyingMouse is moved by one meter to the left the LRU does the same. If the trigger is released, LRU and FlyingMouse are separated and a movement of the FlyingMouse does no longer move the LRU.

Figure 4: FlyingMouse.

With the FlyingMouse it was much easier to control the LRU than with the SpaceMouse. The user just needed to be made familiar with the control method but he required no extensive training. It took several attempts for the test persons to find the right path to the LRU end position. Most of the test persons considered the task very difficult, but possible to be performed. Most problems were caused by the insufficient visualisation of the points where the LRU collided with the aircraft structure, because these points were often obstructed by the LRU itself. Also the limited flexibility of the wrist to perform rotations caused unnatural movements which quickly led to fatigue of the wrist.

The FlyingMouse is a promising method to find mounting paths when the purpose of the application is "design to maintainability". For training purposes, however, we see no advantages over the LRU tracking method (2).

4.0 CONCLUSION

Interaction should always be as simple as possible. It is easier for a user to work within the VR environment and manage the complexity of a particular task if the interaction is intuitive. The selection of the interaction method is essential for the success of a VR application. So before a VR user interface is implemented, the purpose of the user interaction within the specific application must well-defined. It is the basis for the decision on the kind of interaction.

We have seen that the type of VR interaction that copies as exactly as possible the interaction occuring in the real world (see section 2) should only be applied if it is associated with sufficient precision. On the other hand whenever we refrained from reproducing the real interaction method and used available interaction tools (SpaceMouse, FlyingMouse) we experienced that the user had to be trained before he was able to use the tool.

[3] The Cave is stereoscopic projection system that has the shape of a cube. The Cave we used had four projection screens: three side walls and the floor. The dimensions of the Cave were 3m x 3m x 2.4m.

Our studies have also shown, that the interaction method has to be selected very carefully, because an inadequate method can make a solvable task insolvable. This is often the case, when the interaction in the virtual environment is associated with a breach between visual and haptic perception. So if the interaction within a particular application comes along with such a breach, it is better to look for another interaction method, even if it requires user training.

5.0 REFERENCES

[1] RTO-MP-54, "The Capability of Virtual Reality to Meet Military Requirements", Orlando, Florida, USA, 1997.

[2] RTO-MP-058, "What Is Essential for Virtual Reality Systems to Meet Military Human Performance Goals", The Hague, The Netherland, 2000.

[3] Jens Eckstein, "Echtzeitfähige Kollisionserkennung für Virtual Reality Anwendungen", dissertation at Universität des Saarlandes, 1998.

[4] Matthias Buck, "Simulation interaktiv bewegter Objekte mit Hinderniskontakten", dissertation at Universität des Saarlandes, 1999.

[5] http://en.wikipedia.org/wiki/Line-replaceable_unit

[6] "Samira - virtual touch at our fingertips"
http://www.eads.com/web/lang/en/800/content/OF00000000400004/2/80/41142802.html