PC-Based Control Unit for a Head Mounted Operating Microscope for Augmented Reality Visualization in Surgical Navigation

Michael Figl^{1,4}, Wolfgang Birkfellner^{1,2}, Franz Watzinger³, Felix Wanschitz³, Johann Hummel¹, Rudolf Hanel¹, Rolf Ewers³, and Helmar Bergmann^{1,4}

¹ Department of Biomedical Engineering and Physics, AKH, Vienna, Austria

² CARCAS Group, University Hospital of Basle, Switzerland

³ Department of Oral and Maxillofacial Surgery, AKH, Vienna, Austria

⁴ Ludwig-Boltzmann Institute of Nuclear Medicine, Vienna, Austria

Abstract. We have adapted a miniature head mounted operating microscope for AR by integrating two very small computer displays. To calibrate the projection parameters of this so called Varioscope AR we have used Tsai's Algorithm for camera calibration. Connection to a surgical navigation system was performed by defining an open interface to the control unit of the Varioscope AR. The control unit consists of a standard PC with an dual head graphics adapter. We connected this control unit to an computer aided surgery (CAS) system by the TCP/IP interface. In this paper we present the control unit for the HMD and its software design. We tested two different optical tracking systems, the Flashpoint (Image Guided Technologies, Boulder, CO), which provided about 10 frames per second, and the Polaris (Northern Digital, Ontario, Can) which provided at least 30 frames per second, both with a time delay of one frame.

1 Introduction

Due to the fact that computer aided surgery (CAS) was first introduced to the field of neurosurgery, many efforts were focused on introducing Augmented Reality (AR), the overlay of computer-generated graphics for providing "target information" into operating microscopes [3,4,5,6]. and the computer graphics are to be displayed in the optical system of the microscope, therefore no focusing problems when merging the computer graphics and the optical image are encountered. On the other hand, the operating microscope is bulky, expensive, and has a rather limited field of clinical applications. In parallel, head-mounted displays (HMD) were also used for AR in medicine [7,8]. HMDs are lightweight, cheaper, and can be attached directly to the surgeon's head. A major problem, lies in the fact that a common focal plane is not easily achieved in commercial HMDs. Furthermore, rapid head motions cause perceptible delay in the display of the computer graphics, thus causing simulator sickness and reduced acceptance.

Two methods have been proposed in the past to overcome these problems. In [7] and [8], the authors describe video see-through systems. The real world

scenery is being captured by two miniature video cameras, and the augmentation by computer graphics is achieved by mixing the video signals with the computer graphics using a dedicated control unit for the HMD. The user sees the augmented scenery using miniature video monitors. While this allows for strain-free viewing since the complete scene appears focused on the HMDs miniature monitors, the quality of the miniature video monitors is generally poor, and in case of total system failure, the surgeon is blinded. Furthermore, mixing the video signals within a reasonable time span requires a considerable amount of computing power. Another possibility to solve the focusing problem is to move the beam-splitter closer to the scenery. This can be achieved, by using semi-transparent panels [9,1]. The resulting parallax problem can be solved by using an integral photography approach as pointed out in [2]. The resulting displays are, however, somewhat bulky, and have to be placed close to the sterile operating field.

We ended up with developing the Varioscope AR, a prototype of a small head-mounted operating binocular; among the design challenges were the simple integration into existing CAS systems, and the minimization of latencies in the display of the computer graphics. Our solution to these problems is presented in this paper.

2 Materials and Methods

2.1 The Head Mounted Display

The Varioscope AF3, an optical miniature head-mounted operating binocular developed and produced by Life Optics, Vienna (<http://www.lifeoptics.com>), was adapted for AR by adding two miniature LCD displays with VGA resolution (AMEL640.480.24, 640×480 pixels, Planar Microdisplays Inc., Beaverton/OR) with adequate projection optics. We refer to this prototype as the Varioscope AR. The projection optics is designed so that the eye lens of the Varioscope magnifies both the image from the main lens and from the projection optics, both providing an image in the focal plane of the main lens, thus achieving common focus for the optical real world image and the computer graphics. Calibration from world coordinates to display coordinates was established by applying Tsai's camera calibration algorithm [10] for each optical channel separately, thus achieving true stereoscopic vision with parallax correction for the left and right eye. Photos, Details on the design of the Varioscope AR and on the calibration can be found in [11].

2.2 System Setup

The Varioscope AR is designed as an add-on for an arbitrary CAS system with a visualization engine and an optical tracking system. Since assumptions on the performance of a CAS system cannot be made in a strict enough manner we have decided to split the computational task of preoperative planning and visualization on radiological images on the one hand, and the generation of

the augmented reality scenery in the HMD on the other hand. This was performed by an asynchronous TCP/IP connection between the CAS workstation and the control unit for the Varioscope AR. The CAS workstation used for that experimental setup was VISIT, a surgical navigation system developed at our institution [12,13]. VISIT supports two optical tracking systems, the Flashpoint 5000 (Image Guided Technologies, Boulder, CO) and the active Polaris (Northern Digital Inc., Waterloo, Canada). VISIT was adapted in such a manner that it

- acquires position data from the optical tracker through the HMD control unit. This was necessary in order to make sure that a continuous stream of position data from the optical tracker is available to the control unit for rendering the augmentation scenery according to the HMDs actual position independently of the computer load on the CAS system.
- provides raw surgical planning data in the patient coordinate system for generating an OpenGL scenery used for AR visualization in the control unit of the HMD.
- supports a set of commands for additional communication between the CAS workstation and the control unit. The commands can be found in table 1.

Table 1. The commands sent from the CAS system to the control unit of the HMD.

Command	Function
D	send position data
T	read tool list
S	read new scene file
I	read probe IDs
O	read tool offset
X	read data from the tracker

The rendering of the scene, visualized in the HMD, is done by means of OpenGL rendering using the freely available Mesa 3.2 library (<http://www.mesa3d.org>). The interprocess communication of the two processes for visualization on the AR-scenery on the two displays was implemented using System V semaphores. The hardware for the control unit was a standard PC System with an Intel Pentium III Central Processing Unit with 933 MHz, 512 MB RAM, with an Ethernet Network Adapter (3Com 3C905TX) and an ASUS V7100/2V1D graphics adapters with a GeForce 2 MX Graphics Processor running SuSE Linux 7.1 (SuSE GmbH, Nürnberg, Germany).

2.3 Notations and System Preparation

The vector consisting of the coordinates of the point x in the system A is expressed by x_A , the transformation from system A to system B consists of a rotation R_{AB} and a translation T_{AB} , thus: $x_B = R_{AB}x_A + T_{AB}$. We use five

coordinate systems, the world coordinate system, x_W , which is the system of the calibration grid, the camera system, $x_C = (X_c, Y_c, Z_c)^t$ which is the system of a display in the HMD, the source system, x_S , which is the system of a trackable frame rigidly mounted to the calibration grid and the system of the HMD itself, x_H , at least the system of the dynamic reference frame (DRF), x_R , which is mounted on the patient. X_i, Y_i denote the image coordinates, X_f, Y_f the row and column numbers of the point in the display. The constant d is the center to center distance from a pixel to its neighbour.

The calibration of the projection parameters, as described in [11] provides the transformation R_{WC} , T_{WC} , the grid coordinates of the center C_x, C_y and the effective focal length f . Immediately after this camera calibration the tracker is used to provide R_{SH} , T_{SH} . The transformation R_{WS} , T_{WS} is calculated by measuring several points of the system x_W in coordinates of x_S using the tracker. The last transformation R_{HR} , T_{HR} is provided by the tracker during the time the control system is running. From the relations:

$$x_H = R_{HR}^t(x_R - T_{HR}) \quad x_S = R_{SH}^t(x_H - T_{SH}) \quad (1)$$

$$x_W = R_{WS}^t(x_S - T_{WS}) \quad x_C = R_{WC}x_W + T_{WC} \quad (2)$$

$$X_i = f \frac{X_c}{Z_c} \quad Y_i = f \frac{Y_c}{Z_c} \quad (3)$$

$$X_f = \frac{1}{d}X_i + C_x \quad Y_f = \frac{1}{d}Y_i + C_y \quad (4)$$

we derive x_C , as a function of x_R and R_{HR} , T_{HR} . Then X_i, Y_i is the result of a perspective projection, realized by the OpenGL function `glFrustum` with the near clipping plane in the distance f and the far clipping plane in the distance of 800 mm. The row and column numbers X_f, Y_f are calculated by use of the center C_x, C_y this transformation is realized by the function `glViewport`, for information about OpenGL see [14].

The typical image which we want to display consists of surgical planning data which we will refer to as the *scene*, and the surgical instrument tool. The coordinates of the elements of the scene are given in a coordinate system which is defined by surgical mini screws, it is registered to the system x_R of the DRF, by point-to-point registration which provides also the coordinates of the scene in the system x_R , see [13]

2.4 Program

The program consists of three processes, which are executed in the order father, child and grandchild. The fathers main task is the communication with the CAS system and the tracker whereas the child and the grandchild display the two images.

The flow diagram of the main program is shown in the left of figure 1, is contains the following steps for the father process (righthanded course):

- Initialization of the semaphores to achieve the desired order of processes running which is father, child and grandchild.

- Opening a TCP/IP socket and synchronize with the CAS system. At this point the control unit is waiting for the CAS system to continue.
- Opening a pipe to communicate with the child and generating a child process by the `fork` command.
- Having received a command from the CAS system the process has to respond as shown in table 1. Otherwise the default command `X` is assumed and the recent position data, namely R_{HR} , T_{HR} , R_{DR} , T_{DR} is read from the tracker.
- Sending the command of the CAS system together with the position data to the child process. After having made the child process runnable the father process returns to the beginning of its main loop.

The child and grandchild processes are shown in the lower left side of the flow diagram figure 1, the lefthanded course after the first `fork()`, and their glut main loops are shown in the right of fig.1.

- Loading of the calibration data.
- Opening a pipe to communicate with the grandchild and generating a grandchild process by the `fork` command.
- Initializing the glut window, ie starting the OpenGL window and entering the main loop.
- Reading the command and the position data out of the pipe.
- Calculating the minimal distance between the surgical instrument and the planned scene. If this distance is below a previously chosen threshold the scene is painted using wire framed surfaces otherwise it is painted using full surfaces.
- Depending on the command received from the CAS system via the pipe the two processes have to draw the scene using new position data or to load the new scene.
- Command and position data are sent to the grandchild. After having made the succeeding process runnable the processes return to the beginning of their main loops.

3 Results

In an experimental study we attached 16 steel spheres with 4 mm diameter to the base of a skull. They were identified as preoperative planned targets and the x_R coordinates of their locations were sent to the control unit of the HMD. Figure 2b shows the view as seen in the HMD, the image was taken by a camera mounted on the left eyepiece. The Flashpoint 5000 tracking system flashes the IR-LEDs separately one after the other with a maximum frequency of about 300 IR-LEDs per second. Therefore we expected to get lower frame rates for higher numbers of LEDs, whereas the Polaris system illuminates the LEDs all at the same time. We achieved at least 30 frames per second using the Polaris tracker, and no more than 18.5 with the Flashpoint tracker. The results are described in table 2, the framerate is the average framerate of 10 measurements, each having

Fig. 1. Flow diagram of the main program and the child's main loop. The father process communicates with the CAS system and the tracker. The child and the grandchild process display the two images.

Fig. 2. a) The skull as seen through the Varioscope AR before augmentation is switched on. b) The augmented scenery as seen through the Varioscope AR. The steel spheres are overlaid by their virtual counterparts on CT. A needle-shaped surgical instrument approaches from the right.

a duration of 10 seconds. It is evident that the framerate varies considerably when using the Polaris tracker. The reason for this behavior is not completely clear, we assume a synchronization problem in the tracker's internal hardware to be the source.

Table 2. The framerate using the two different optical trackers with different numbers of tools and LEDs. σ denotes the standard deviation

Polaris Tools	Framerate	σ	Flashpoint Tools	Framerate	σ
3 Tools, 11 LEDs	45.33	10.18	3 Tools, 19 LEDs	13.45	0.05
3 Tools, 12 LEDs	40.35	11.45	3 Tools, 14 LEDs	17.71	0.03
2 Tools, 8 LEDs	33.54	3.52	3 Tools, 8 LEDs	18.37	0.05
2 Tools, 7 LEDs	32.8	2.62	2 Tools, 6 LEDs	18.5	0.01

4 Discussion and Conclusions

We believe that the future of CAS depends on two factors. First, the development of satisfying solutions for a wide variety of clinical specialties beyond the current classic application fields is to be fostered. Second, the intuitive man-machine interface for these solutions has to reach a stage where the CAS-system becomes an 'embedded' system in the sense that it is no longer a dominant additional high-tech apparatus in the operating room but a simple, easy-to-use piece of equipment providing an aid to the surgeon in clinical routine use.

AR will be a part of this development for providing an intuitive visual feedback on surgical progress, provided that interfacing such an AR system with sufficient performance concerning image quality, ease of use, and latency is possible for existing CAS infrastructure at reasonable cost. The possible performance of such an approach for visualization in the Varioscope AR is documented in this paper. With a simple off-the-shelf computer, stereoscopic visualization is achieved at frame rates in the range of 40 Hz; due to the sophisticated structure of the HMD-control software, the only limiting factor for even increased visualization speed is the tracking system. The fact that all information necessary (mainly the OpenGL-scenery in patient space provided by the external CAS system, and the optical tracker data provided by the HMD control system) is shared using a standard TCP/IP connection makes the integration of such a system into an arbitrary navigation system easy.

Acknowledgment

The authors wish to thank Dr. M. Lehl and her colleagues at Life Optics, for the close cooperation and the excellent relationship. This work was supported by the Austrian Science Foundation FWF under research grant P 12464 MED.

References

1. M. Blackwell, C. Nikou, A. M. DiGioia 3rd, T. Kanade: "An image overlay system for medical data visualization", *Med Image Anal* 4(1), 67-72, (2000).
2. S. Nakajima, K. Nakamura, K. Masamune, I. Sakuma, T. Dohi: "Three-dimensional medical imaging display with computer-generated integral photography", *Comput Med Imaging Graph* 25(3), 235-241, (2001).

3. D. W. Roberts, J. W. Strohbehn, J. F. Hatch, W. Murray, H. Kettenberger: "A frameless stereotaxic integration of computerized tomographic imaging and the operating microscope", *J Neurosurg* 65(4), 545-549, (1986).
4. D. W. Roberts, T. Nakajima, B. Brodwater, J. Pavlidis, E. Friets, E. Fagan, A. Hartov, J. Strohbehn: "Further development and clinical application of the stereotactic operating microscope", *Stereotact Funct Neurosurg* 58(1-4), 114-117, (1992).
5. R. Shahidi, R. Mezrich, D. Silver: "Proposed simulation of volumetric image navigation using a surgical microscope" *J Image Guid Surg* 1(5), 249-265, (1995).
6. P. J. Edwards, A. P. King, C. R. Maurer Jr, D. A. de Cunha, D. J. Hawkes, D. L. Hill, R. P. Gaston, M. R. Fenlon, A. Jusczyck, A. J. Strong, C. L. Chandler, M. J. Gleeson: "Design and evaluation of a system for microscope-assisted guided interventions (MAGI)", *IEEE Trans Med Imaging* 19(11), 1082-1093, (2000).
7. H. Fuchs, M. A. Livingston, R. Raskar, et al.: "Augmented Reality Visualization for Laparoscopic Surgery", in W. M. Wells, A. Colchester, S. Delp (eds.): "Medical Image Computing and Computer-Assisted Intervention - MICCAI'98", Springer LNCS 1496, 934 pp., (1998).
8. C. Maurer Jr, F. Sauer, B. Huc, B. Basile, B. Geiger, F. Wenzel, F. Recchi, T. Rohlfing, C. Brown, R. Bakos, R. Maciunas, A. Bani-Hashemi: "Augmented reality visualization of brain structures with stereo and kinetic depth cues: System description and initial evaluation with head phantom" in *Medical Imaging 2001: Visualization, Display and Image-Guided Procedures*, Seong Ki Mun, Editor, Proceedings of the SPIE Vol 4319, 445-456 (2001)
9. M. Blackwell, F. Morgan, A. M. DiGioia 3rd: "Augmented reality and its future in orthopaedics" *Clin Orthop* 354, 111-122, (1998).
10. R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses", *IEEE Trans Robot Automat* RA-3(4), 323-344, (1987).
11. W. Birkfellner, M. Figl, K. Huber, J. Hummel, R. Hanel, P. Homolka, F. Watzinger, F. Wanschitz, R. Ewers, H. Bergmann: "Calibration of projection parameters in the Varioscope AR, a head-mounted display for augmented reality visualization in image-guided therapy", in *Medical Imaging 2000: Display, and Image-Guided Procedures*, Seong Ki Mun, Editor, Proceedings of SPIE Vol. 4319, 471-480(2001).
12. W. Birkfellner, K. Huber, A. Larson, D. Hanson, M. Diemling, P. Homolka, H. Bergmann: "A modular software system for computer-aided surgery and its first application in oral implantology", *IEEE Trans Med Imaging* 19(6), 616-620, (2000).
13. W. Birkfellner, P. Solar, A. Gahleitner, K. Huber, F. Kainberger, J. Kettenbach, P. Homolka, M. Diemling, G. Watzek, H. Bergmann: "In-vitro assessment of a registration protocol for image guided implant dentistry", *Clin Oral Implants Res* 12(1), 69-78, (2001).
14. OpenGL Architecture Review Board, "OpenGL Reference Manual - Second Edition", Editors: R. Kempf, C. Frazier, Addison-Wesley, (1997).