VRIC, Virtual Reality International Conference, Laval Virtual 2001, May 16-18

Augmented Reality Collaborative Environment : Calibration and Interactive Scene Editing

Raphaël Grasset, Xavier Decoret and Jean-Dominique Gascuel
IMAGIS - GRAVIR/IMAG - INRIA
655, av. de l'Europe 38334 Montbonnot CEDEX
iMAGIS is a joint project between CNRS, INRIA, INPG, and UJF
{Raphael.Grasset|Xavier.Decoret|Jean-Dominique.Gascuel}@imag.fr

Keywords: Augmented Reality, Calibration, Registration, Tangible User Interface

Abstract: This paper presents a innovative environment for collaborative applications in augmented reality. This environment is based on a table-top metaphor where each user, sitting around a table, is head and hand tracked. See-through head mounted devices are used to display virtual objects, as if they were lying on the table, together with real objects. This approach requires accurate calibration and registration techniques as well as an intuitive user interface in order to allow a realistic combination of real and virtual objects.
We focused on the easiness of the calibration procedure and present a new method that is both very simple to setup and very efficient. This method is used to calibrate together the head mounted display, the hand stylus, the shared interaction space, and the tracking system. It achieves sufficient accuracy for non-critical applications such as product design, gaming, etc…
Furthermore, we describe a new method to dynamically add real objects into the system, providing a unified treatment of virtual and real objects. This method can deal with a larger range of objects than existing methods.

1. Introduction

In both professional and social life, we need to collaborate with other people : meeting between engineers and researchers for the design of a new product, planning of a new building between architects, board-game session between friends [21], etc… In these situations, people often sit around a table that provides a common surface for both verbal and non verbal communication (such as gesture and facial expression [10]). This surface is also used to place objects that enhance the communication (game pawns, artworks) or which have to be discussed (plans, prototypes of the designed objects, etc…).

Fig 1. Virtual objects place on a real board.

1.1. Virtual Environment vs. Augmented Reality

Virtual environments (VE) can be developed to provide computer assistance for such collaborative activities. However, in these environments, the user is completely immersed, therefore isolated from the real world and collaborators. All subtle communications facts must be rendered by the system in the virtual environment else they will not be available to the user. Moreover, interaction with the world is difficult and not always intuitive. We found this drawbacks to be critical for the kind of application we are considering.

In contrast, Augmented Reality (AR) is a good medium for these activities. Indeed, it does not suppress natural communication (the users can still see each other) but enhances communication by providing new tools that mimic the natural interaction metaphor.

Augmented Reality has some other advantages:

- computer graphics can be used to add animations, visual effects or visual clues

such as display of hidden faces or objects' interiors, use of false colors, sketching, etc…

- computers provides up-to-date digital data (where, for example, a traditional architect's artwork take some times to be updated after some modifications have been decided);
- computers can afford complex computations thus reducing the gap between physical and digital data;
- the use of one head mounted display (HMD) per user allow to display user specific data, thus providing us with the concepts of private and dedicated views;
- new interaction metaphors can be used such as hand gesture recognition (using computer vision), virtual manipulators ([23]).

With Augmented Reality, we can imagine the following scenario for a collaborative session. Participants are engineers. The goal of the session is to design the loading procedure of manufactured products onto a train. The computer simulates the flow of items conveyed by mechanical units from the storage place to the wagons and displays an animation about on the table[1]. All participants comment what they see, making gestures, looking at each others. They naturally point out the stage of the process they are talking about and the system will highlight them, maybe freezing the animation for a while. A few engineers gather on the corner of the table to discuss privately a minor point, joining back the main discussion a few minutes later. In order to change the path of a mechanical unit, one engineer introduces a plastic cube, places it on the table at the same location where the unit is displayed, and tells the computer to bind these 2 objects. Every-one can now move the cube to alter the position of the virtual unit. The simulation is accordingly adapted by the computer. All engineers view the virtual objects with sketched information matching their concerns : the security specialist sees moving units colored according to their speed and the weight they are carrying while the product manager sees colored items matching the train they should be loaded on.

1.2. Requirements

In such a scenario, the key points are a perfect integration of real and virtual objects (as stated by Kato[13] for his enhanced shared space application), as well as a natural interaction procedure. We developed our system to met the following requirements :

- tracking and registration accuracy;
- efficient and non intrusive command system;
- dynamic modification of the virtual database.

1.3. Overview

In section 2, we first review previous works related to our approach. Section 3 gives a general description of our table-top system and how it is setup. Section 4 gives the details of the calibration technique we developed. We then present the user procedure to dynamically insert both virtual and real objects into the system (section 5). We then conclude and open the discussion on future works.

2. Related Work

2.1. Collaborative Environment

One of the first collaborative application is the AR2 Hockey project ([17]). This system presents a real-time air-hockey game for two players equipped with HMD. Later works by Agrawala[1] or Szalavari[24] developed this system style of interaction. Tangible User Interface, recently coined by Ishii[12], is a general metaphor that use tracked real objects as manipulators of virtual ones. Similar ideas can also be found in previous or contemporary works[19, 8, 13]. This paper is a continuation of the master thesis of [11], and can also to be related to problematic explored by [6].

2.2. Registration

Registration's concern is the alignment between virtual and real elements. Azuma [2] classified the possible alignment errors in two categories :

- **Static error** defined as the misalignment the user can perceive when he is immobile;
- **Dynamic error** defined as the misalignment the user can perceive when he moves.

Static error is due to various causes : position error of some trackers, noise, numerical errors, tracker's non linearity, etc… **Calibration** is the process or compensating all those error sources.

Dynamic error is due to the time lag between objects or user's movements and visual feedback (latency of acquisition and signal processing) and the main difficulty is to compensate this lag. As in many systems, we used Kalman filtering to reduce both noise and time lag to acceptable levels.

There are many work on calibration : [2,22,25]. One can refer to [3] for an exhaustive review. For our concerns, we only focused on interactive calibration methods for optical see-through HMD, reviewing their simplicity versus their accuracy.

[1] That look like if it is on the table through every user personnal display.

Fig2. Virtual objects occlude by a real object.

Azuma[2], and Oishi[16] proposed methods that achieve high accuracy. But they require many technical steps and extra hardware devices which makes the system demanding on time and concentration for the user. More recently, Genc[26] proposed a more intuitive method for the camera calibration, where the user does not need to have a static posture during the calibration. He is asked to match, by moving the head, a virtual point on the HMD and a reference world point. It also achieves very good accuracy but still requires several stages (at least six point must be matched) and some skill from the user to align the points. Fuhrmann[9] proposes a similar method where the reference point is no longer static but tracked with a hand-held stylus, which facilitates the matching procedure.
We will see in section 4 how we can trade off some accuracy for a single stage easy calibration procedure.

2.3. Interaction with real object

Interacting with real objects issues are :

- **occlusion compatibility** between virtual and real objects: parts of virtual objects behind by real objects should not be displayed;
- **respect of laws of physics** between the two worlds: virtual object laid on top of real objects should remain in contact;
- **manipulation awareness** : the system must be updated after a real object is moved in the environment.

Real objects therefore need to be acquired by the system. This involves specifying :

- their geometry;
- their appearances;
- their positions (rotation, translation) in the world coordinate system (CS).

This can be acquired in different ways.
It can be reconstructed using techniques borrowed from research in Computer Vision or even scanned using an (expensive) 3D scanner. Both solutions requires lengthy computations are not suitable for interactive sessions.

The object can also be interactively acquired by specifying key points with a tracked stylus either to build the object from scratch or to place (e.g. computing a transformation matrix) reference model[25]. The choice of the point depends of the object to be build or matched, which makes the calibration object dependent.

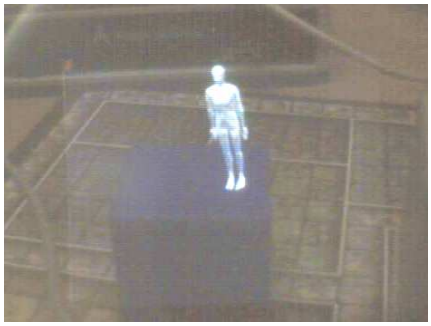

Fig3. Virtual object place on a real cube.

In the same spirit Baillot[4] and Fuhrmann[9] propose manual methods to specify the transformation between a virtual representation of an object and its real incarnation. These methods can be very long due to the number of parameters to specify.
Lee[14] described an interesting modeling application in which geometry is acquired through user specified key points whereas appearance textures are acquired through a vision system. They demonstrate their results on complex shapes like a Japanese teapot.
After it has been thus added, the real object may be moved and its position then needs to be tracked. Pure vision methods exist[15] but require a lot of computational power to achieve real-time 6 DOF tracking. A practical solution currently used [5, 22] is to place a noticeable feature on the object to facilitate vision techniques, and to use constrained systems (e.g. objects must lie on the table).
We use a brute force, easy to implement, approach where objects allowed to be moved are attached a tracker. But we plan to investigate more subtle and versatile methods, inspired by [7].

3. The multi-user table-top setup

We use a centralized configuration based on around-table configuration. Every user is sited around a table. This solution allows face to face collaboration, and maintain easy gesture or verbal communication. We believe that a sitting position (opposed to stand-up as in other system such as the Studierstube or Shared- Space) is more realistic for long period of use.
This configuration moreover matches the one naturally adopted when playing multi- layers game like Trivial Pursuit, Monopoly, or any board-game. We besides decided to use a real board coming from the Heroquest game to decorate our workspace : it

defines and materializes the shared arena for all people interacting within the system. We can also

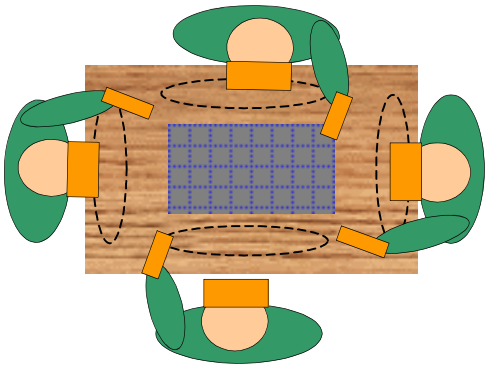

Fig4. Target configuration with 4 users. Each user have a tracked HMD and a tracked stylus for the interaction. We defined private and shared zone in the central area.

take advantage of the regular grid drawn on such boards to facilitate accurate placement of objects.
Every user is equipped with an optical see-through HMD and have a stylus for interaction. We envisage that a specific user (generally the session's leader, known as game master in the Heroquest-and-like terminology) may have a laptop for more complex interaction.The reasons why we adopted this visualization mode are :

- it preserves the direct view between users;
- it allows display of user specific informations (private views) which cannot be done with projectors or active display table (such as responsive workbench or virtual workbench);
- its low weight, and the easy installation procedure let users quickly start a session.

Fig5. Hardware use for the developed prototype. We can notice a HMD I-glasses, a pen, and the Flock of Birds systems (an emitter, 2 receptors, and control unit).

We choose an optical see-through rather than a video see-through model to avoid extra costly computations and the resulting video latency which we found very disturbing (conflicts between perception and action). The HMD we currently use are i-glasses with stereoscopic mode.

For the head and hand tracking we currently use a magnetic tracker system. Despite of its noise measurement and wiring this system have fast response time and no line of sight limitation. For interaction we choose a stylus with the magnetic tracker attached on it (grabbing interface). This interface responds to quick and complex movements with low computation time (as opposed to hand gesture recognition with vision tracking systems for example).

We choose a centralized shared-memory system in order to focus on the testing of our system (usability, ergonomy, robustness) rather than development of technical solutions to distributed synchronization and network programming problems. This choice was facilitated by the presence in our laboratory of an Origin 2000 with 6 processors, 1 Infinity Reality and 4Go of shared memory.

We developed using C++ and the Performer API[18].

4. A fast user calibration technique

We want to have a calibration procedure that require the minimum effort to be as close as possible to some ideal “plug’n play” system.
This is mainly required by two situations : the HMD is shared among people (that are morphologically different), or people bring they’re own tracked-HMD system (that should then be calibrated against the common tracker).

4.1 Static calibration

Many coordinates systems (CS) co-exist as shown by figure 6. We used the taxonomy proposed by Robinett[20] and summarized in table 1. To work with these different CS, we need to know the transformation matrix (noted T_B^A) that maps a coordinate system A to a coordinate system B.

HMD

Image CS	pixel coordinates in the projected image
Eye CS	located at the eyes (a left eye CS and a right eye CS for stereo)
Head CS	centered in between the eyes
Receptor CS	used by the tracker fixed on the HMD

Board

World CS	attached to the board-game
Emitter CS	specific for each player

Stylus

Transmitter CS	use by the tracker fixed on the pen
Pen CS	located at the tip of the pen (used for interaction with environment)

Table1. Coordinate systems

The transformation between CS of a same group (board, Stylus, HMD in table 1) are fixed and need only to be computed once, whereas transformations between CS across different groups will be dynamically retrieved from the trackers.

Calibration is the process to accurately compute these static transformation matrices for each group.

HMD Calibration

In this process, we use the HMD's intrinsic parameters : horizontal and vertical FOV, focal distance, image resolution, distance between eye (De). We obtain these informations from the HMD constructor as a perspective projection matrix noted T_{persp}^{ideal} and a value for De.

Since we assume the use large volume of commercial HMD (as opposed to unique research prototypes), we consider this information to be always available:

Because one user's *inter pupillar distance* (IPD) does not generally match the HMD eye distance, we need to correct the ideal matrix to obtain an off-axis perspective projection matrix T_{persp} for each eye. We assume that the user knows his IPD (it can be measured easily) and "carries" it with him when he plans to use our system.

The only transformation we need is $T_{receptor}^{image}$. It is obtained by computing the right hand terms of equation (1) :

$$T_{receptor}^{image} = T_{eye}^{image} \times T_{head}^{eye} \times T_{receptor}^{head} \quad (1)$$

T_{eye}^{image} is simply T_{persp}

and T_{head}^{eye} is a translation of $\pm\frac{1}{2}IPD$

To get $T_{receptor}^{head}$, we ask the user to match a virtual representation of the emitter[2] with the real emitter. Initially, the view in the HMD and the real object does not match. At one moment, the user freezes

[2] This latter being box shaped we use a cube.

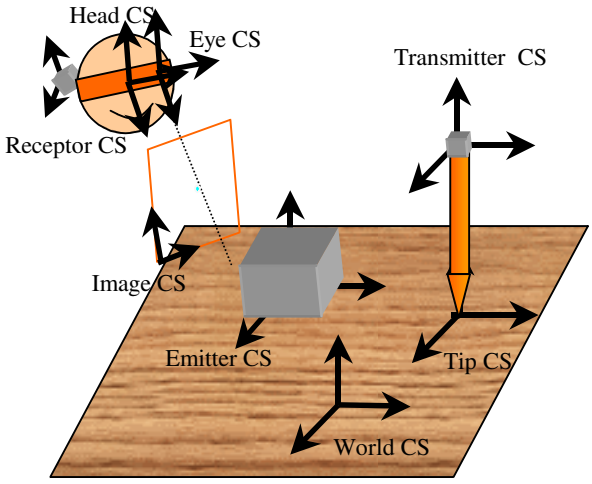

Fig 6. Coordinate systems of our augmented reality system.

the image and moves the head to align them. He then validates with a “click". Since better results will be obtained when the user has to match 4 non coplanar points on the emitter, it is recommended to freeze the view in a non degenerate (axis-aligned) configuration.

We note X the coordinate of a point in the world CS and x the coordinates of this point in the image CS. We have :

$$x = T_{persp} \times T_{world}^{eye} \times X \quad (2)$$

and

$$T_{world}^{eye} = T_{head}^{eye} \times T_{receptor}^{head} \times T_{emitter}^{receptor} \times T_{world}^{emitter} \quad (3)$$

Since we fixed x (the image is frozen) and X (the real point does not move), we can deduce T_{world}^{eye} from (3). We can arbitrarily choose $T_{world}^{emitter} = Id$. At every moment, $T_{emitter}^{receptor}$ is given by the tracking system. Therefore when the user validates the alignment, we can compute $T_{receptor}^{head}$ with :

$$T_{receptor}^{head} = T_{head}^{eye\ -1} \times T_{world}^{eye} \times T_{emitter}^{receptor\ -1} \quad (4)$$

Fig7. Two illustrations before and after the calibration phase. In the second image we can see the virtual emitter register with the real emitter.

We would like to point out that this determination of the transformation matrix is incremental. Though a trained subject easily achieves very good calibration in one step, the system allows the user to incrementally refine the initial result. He can repeat several times the procedure, and the results are iteratively averaged. This matching procedure is also facilitated by the stereo vision.

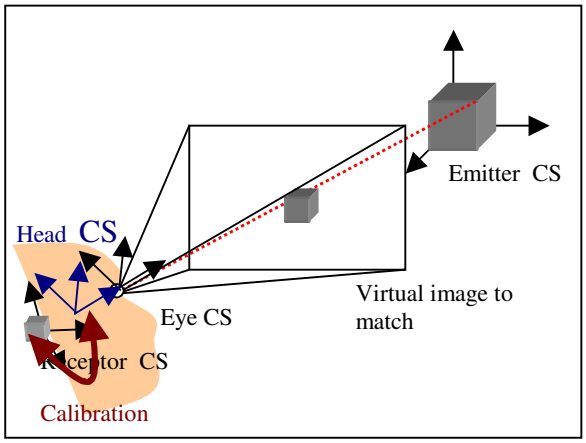

Fig8. The HMD Calibration for estimate the transformation between Receptor CS and Head CS. We can see the user align the freeze view of the virtual emitter with the real emitter.

The error introduced by this calibration procedure is to the order of 0.2 cm at 30 cm of emitter, 0.5 cm at 60 cm of the emitter (working distance), 1.5 cm of accuracy in 70 cm of emitter. Greater distance are meaningless due to increasing level of magnetic noise.

Pen Calibration

The principle to compute the transformation between the transmitter and the tip is to record the stylus position with the tip fixed and different orientations of the body.

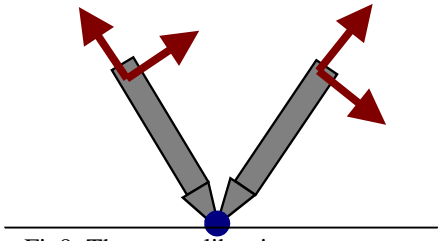

Fig9. The pen calibration.

Usual methods record a lot of positions and perform various kind of filtering. We evaluated these methods and found that due to the noise, the gain in accuracy is not worth the extra measures.
Keeping in mind our objective of simplicity and rapidity for the calibration procedure, we choose to ask the user to roughly orient the stylus in only 4 representatives directions : east, west, north, south (intuitive). The system is then easily solved with a SVD decomposition approach (low computation time).
The error we have is of the order of 0.5 cm.

Fig10. Result of pen calibration.

World Calibration

The user clicks on the 4 corners of the board with the calibrated stylus and the system deduces the transformation between world and transmitter CS (similar to [25]).
Combining the different error, we have inaccuracy of 0.5 to 1 cm at a distance of 50 cm of the emitter, and 2 cm at 70 cm. 50-70 cm is the typical range of use in the application we tested and we found this inaccuracy to be largely acceptable for most practical collaborative application.

Fig11. Different viewpoint of a virtual object (the character stay in the same square).

5. Dynamic objects

5.1. Adding object

We propose an intuitive method based on simple manipulations performed by the user. There are 2 different manipulations depending of the knowledge the system has about the object. It can either be an object for which an equivalent 3D model is available. We refer to it as *a digitalized objects*, though the model could have been produced by an artist and not necessarily scanned. Or it can be an object whose shape is not exactly available but belongs to a generic category known by the system. We call such objects *generic geometric models*. The introduction of this category is motivated by the observation that many objects users of our system would be tempted to manipulate (i.e that usually lie on a desktop) have similar simple shapes. Book, CDs, videotape are box-shaped; bottles, pens, mugs are almost cylinders and paper sheets are rectan-

gles. Moreover this shape can be fully described by a few reference point.

Digitalized models

We need to find the transformation between the object CS and the world CS. To do so, the user is asked to superpose a virtual representation of the model with the real object, with the help of the stylus. We use a "gravity constraint" : because the new object have to be laying on something (already digitized) the problem is only reduced to 3 unknowns (x,y positions, and θ orientation).

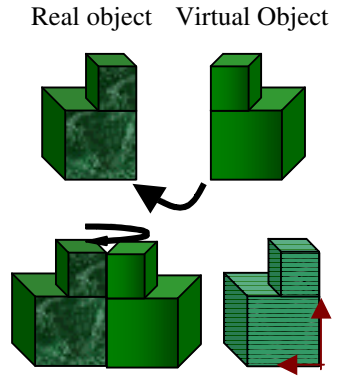

Fig12. Description of our method to add digitalized objects.

In a first step, the system displays the virtual model at the tip of the stylus, as if attached to it by a point P. The user first places this point P to its equivalent P' on the real object, by bringing the tip of the stylus in contact with it. He indicates when he is done with a "click". In the second step, the user changes the orientation of the virtual objects by moving the stylus, until virtual and real matches, which he indicates with a second "click".
The stereoscopic view facilitates this procedure and we obtain surprisingly good result (error of 0.3 to 0.5 cm, that is mainly due to the HMD and stylus calibration errors).

Fig13. Adding a digitalized model.

Generic Geometric models

We use a Canoma-like approach where the user first select a generic shape among the ones available in the system, and indicates with the tip of the stylus the 3D coordinates of the reference point associated to this shape (for example 3 corners for a rectangle).

Our system currently offers the cylinder, box and rectangle shapes but obviously, any generic shape can be added to the system as long as the number of reference points it requires is not too big (or adding new object procedure will be tedious).

Fig14. Adding a rectangle shape.

5.2. Moving objects

We propose a versatile way to move objects based on the observation that the kind of objects we add do not move "by themselves".
We **temporarily** attach a magnetic tracker to the object that has previously been added as described in section 5.1. Since the tracker and object positions are known in world CS, we can compute the transformation matrix between them. The real object can now be moved in any fashion, its position is tracked by the system.
A great advantage of this approach is that the tracker can be fixed in a versatile manner (for instance a magnetic tracker glued with mastic, or an optic tracker printed on a sticker). And the same tracker can be used to move different objects and passed among users.

5.3. Adding virtual object

Virtual objects can be added in exactly the same manner than real objects except that there is no need to make them match with anything. The same set of constraints can be applied to ease the placement, such as maintaining contact surface with the table. This is exactly the same approach than the one used for the PIP ([23]).

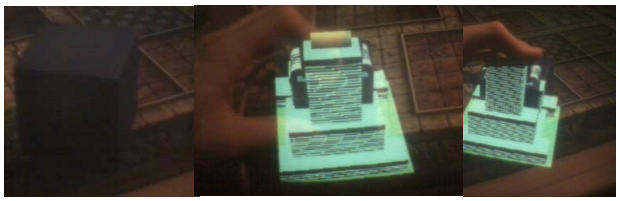

Fig15. Moving a virtual building with a real cube.

6. Conclusion and future works

We proposed augmented reality system dedicated to the kind of collaborative applications where users meet around a table. For this system, we developed new solutions to classic problems that emphasized the easiness and rapidity of use.

We first described a calibration technique that is very simple and intuitive, at the cost of a loss in accuracy. However, the achieved accuracy is stilled perfectly sufficient for a large range of non-critical applications like architectural design, gaming, planning simulation.

We then propose a new procedure to add real objects almost as easily as virtual objects are added in systems like the Personal Interaction Panel. The range of objects that can be added is very large since parametric generic shapes are used. We also proposed a versatile metaphor to move real object.

In the future, we expect to test our system with several users (currently, we have only one HMD+stylus). We also want to reduce the sensibility to noise and the latency of the tracking systems.

References

1. M. Agrawala, A. C. Beers, B. Fröhlich, I. McDowall P. Hanrahan, and M. Bolas. *The Two-User Responsive Workbench: Support for Collaboration Through Individual Views of a Shared Space.* Computer Graphics Proceedings, Annual Conference Series SIGGRAPH '97, 1997, pp. 327-332.
2. R. Azuma and G. Bishop. *Improving static and dynamic registration in an optical see-through display*. Computer Graphics, July 1994, pp. 194--204.
3. R. Azuma. *A Survey of Augmented Reality*. in "Presence: Teleoperators and Virtual Environment 6, 4, August 1997,pp 355-385.
4. Y Baillot *First Implementation of the Virtual Reality Dynamic Anatomy (VRDA) Tool* Master thesis dissertation, School of Computer Sciences. University of Central Florida. January 1999.
5. M. Billinghurst, S. Weghorst and T. Furness. "*Shared Space: An Augmented Reality Approach for Computer Supported Collaborative work*". University of Washington, 1997.
6. M. Broll, E. Meier and T Schardt : *The Virtual Round Table - a Collaborative Augmented Multi-User Environment* in CVE 2000, Churchill, Elizabeth.
7. W. Bruns, B. Volker *Bridging the Gap between Real and Virtual Modeling - A New Approach to Human-Computer Interaction* in Proceedings of the 2nd IFIP 5.10 Workshop on Virtual Prototyping, Arlington Tx, May 1996.
8. A. Butz, T. Hllerer, S. Feiner, B. MacIntyre, C. Beshers. *Enveloping Computers and Users in a Collaborative 3D Augmented Reality*. Proc. IWAR `99, 1999.
9. A. Fuhrmann, D. Schmalstieg and W. Purgathofer: *Fast Calibration for Augmented Reality* Proceedings of ACM Virtual Reality Software & Technology '99 (VRST'99), short paper, London, December 1999.
10. A. Guye-Vuillieme, T.K.Capin, I.S.Pandzic, N.Magnenat Thalmann and D.Thalmann, *Non-verbal Communication Interface for Collaborative Virtual Environments.* The Virtual Reality Journal, Springer, 1999 Vol. 4, pp.49-59.
11. R. Grasset *Realite Augmentee : vers les jeux videos.* Master Thesis Dissertation, Institut National Polytechnique de Grenoble, Grenoble. France. 2000.
12. H. Ishii and B. Ullmer. *Tangible bits: Towards seamless interfaces between people*, bits and atoms. In Proceedings of ACM CHI 97 Conference on Human Factors in Computing Systems, volume 1 of PAPERS: Beyond the Desktop, 1997, pp 234--241.
13. H.Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, K. Tachibana.*Virtual Object Manipulation on a Table-Top AR Environment.* In Proceedings of ISAR 2000, October 2000.
14. J. Lee, G. Hirota and A. State *Modeling Real Objects Using Video See-through Augmented Reality* International Symposium on Mixed Reality Mar 2001.
15. E. Marchand, P. Bouthemy, F. Chaumette, V. Moreau. - *Robust real-time visual tracking using a 2D-3D model-based approach.* - IEEE Int. Conf. on Computer Vision, ICCV'99 , Kerkira, Greece, September 1999.
16. T. Oishi, and T. Susumu *Methods to calibrate projection transformation parameters for see through head-mounted displays*. Presence 5:1. 1996.
17. T. Ohshima, K. Satoh, H. Yamamoto, and H. Tamura: *AR2 Hockey system : A collaborative mixed reality system*, Trans. VRSJ, vol.3, no.2, 1998, pp. 55-60.
18. J. Rohlf and J, Helman. *Iris performer: A high performance multiprocessing toolkit for real-time 3D graphics*. Proceedings of SIGGRAPH`94 (Orlando, Florida, July 24-29, 1994). In Computer Graphics Proceedings, Annual Conference Series, 1994, ACM SIGGRAPH, pp 381-394.
19. M. Rauterberg, M. Fjeld, H. Krueger, M. Bichsel, U. Leonhard, M. Meier. *BUILD-IT: A computer Vision-based Interaction Technique for a Planning Tool*. In (H. Thimbleby, B. O'Connaill, P. Thomas eds.) People and Computers XII, Proc. of HCI'97. Berlin, Springer 1997, pp. 303-314.
20. W. Robinett, and R Holloway. *The visual display transformation for virtual reality*. Presence 4:1.1995.
21. S. D. Scott, G. B. D. Shoemaker and K. M. Inkpen. *Towards seamless support of natural collaborative interactions*. In Proceedings of Graphics Interface. Montreal, May 2000, pp. 103-110.
22. A. State, H. Gentaro, D. T. Chen, W. F. Garrett, and M. A. Livingston. *Superior Augmented-Reality Registration by Integrating Landmark Tracking and Magnetic Tracking*. Proceedings of SIGGRAPH 96 (New Orleans, LA, August 4-9, 1996). In *Computer Graphics* Proceedings, Annual Conference Series 1996, ACM SIGGRAPH, pp. 429-438.
23. Z. Szalavari and M. Gervautz. *The personal interaction panel --- A two-handed interface for augmented reality*. Computer Graphics Forum, 16(3), September 1997, pp. 335-346.
24. Z. Szalavari, D. Schmalstieg, A. Fuhrmann, and M. Gervautz. *Studierstube: An environment for collaboration in augmented reality*. Virtual Reality, 3(1), 1998, pp. 37-48.
25. M. Tuceryan, D. Greer, R. Whitaker, D. Breen, E. Rose C. Crampton, and K. Ahlers. *Calibration Requirements and Procedures for Augmented Reality*. IEEE Transactions on Visualization and Computer Graphics, 1(3), September 1995, pp. 255-273.
26. Y. Genc, F. Sauer, F. Wenzel, M. Tuceryan, and N. Navab, *Optical See-Through HMD Calibration: A Novel Stereo Method Validated with a Video See-Through System* in Proceedings of the IEEE and ACM International Symposium on Augmented Reality, Munich, October 2000. pp. 165-174.