Line of Sight Method for Tracker Calibration in Projection-Based VR Systems

Marek Czernuszenko, Daniel Sandin, Thomas DeFanti
{marek | dan | tom} @evl.uic.edu
Electronic Visualization Laboratory (EVL)
Department of Electrical Engineering and Computer Science
and School of Art and Design
University of Illinois at Chicago
851 S. Morgan, room 1120 SEO
Chicago, IL 60607
(312) 996-3002

Abstract

This paper describes a method for correcting static errors in the position component of a 6-degree-of-freedom tracker in a projection-based VR system. This method allows users to observe where errors in the environment are significant and correct them interactively. Later touch-up correction is possible as well. This technique is based on superimposing targets in physical space with their virtual images. The only hardware addition to the VR system required is a few precisely placed targets.

Keywords: Tracker calibration, Virtual Reality, Projection-Based VR, CAVE, ImmersaDesk.

1 Introduction

1.1 Motivation

Six-degree-of-freedom (6DOF) trackers are widely used in VR systems. Computer graphics systems need information about the location of the user's eyes to generate an image of the scene from the correct, user centered perspective. At the same time the location of an input device like a glove or a wand is needed to enable user interactions with a virtual environment. Widely used electro-magnetic trackers are sensitive to electrically or magnetically conductive objects in the environment. Static errors as high as 40% (4 feet) have been observed near the maximum range of the tracker [Ghazisaedy et al., 1995]. These errors are not acceptable for many applications. Two approaches of correcting these static errors are described by Bryson [Bryson, 1992] and Ghazisaedy [Ghazisaedy et al., 1995]. Both of these methods require precision placement of the receiving sensor in a large number of positions. In the case of the CAVE [Cruz-Neira et al., 1992], which is 10x10x10 foot^3, 1000 measurements are required to obtain a full 1 foot interval correction table. In practice not all areas are reachable by user, so 400 measurements are enough to calibrate the CAVE [Ghazisaedy et al., 1995]. More measurements are required for a finer calibration table. Because this process involves so many precise measurements of 3D location, the process of gathering data is time consuming.

In this paper we present a method which is based on the user moving around the space and aligning real and virtual objects where correction appears to be needed. This method avoids the repetitive measurement of positions in 3D space and reduces the number of correction points required by allowing the user to concentrate on the area of the space which most needs corrections.

1.2 Errors in Superimposition of Real and Virtual Objects

Our procedure could be applied to any tracker used in a projection-based VR system. These VR systems include the CAVE^{tm}, ImmersaDesk^{tm} [1] [Czernuszenko et al., 1997], Infinity-Wall [Czernuszenko et al., 1997], Responsive Workbench [Krueger and Froehlich, 1994] and fish-tank VR. In these systems the user wears lightweight LCD shutter glasses. Images are projected on large screens or they are viewed on a monitor. Information about the user's eye locations is obtained from a tracker system. A 6DOF tracker reports positional and orientational data of receivers in its own coordinate system. This data contains static and dynamic errors. Our system compensates for static errors by employing a correction lookup table.

The correction table is a uniform 3D array of point pairs. The first point is a reported tracker position and second is a corresponding corrected position. Based on this lookup table any location reported by the tracker can be corrected by interpolating between a few points from the table. This technique was described by Bryson [Bryson, 1992] and Ghazisaedy [Ghazisaedy et al., 1995]. Corrected tracker readings are transformed to a World Coordinate System (WCS) which is used by the application (Figure 1).

In projection-based VR systems, the user sees both virtual and real objects. For example, the user holding a wand sees the physical wand, as well as a drawing of that device (provided that the application

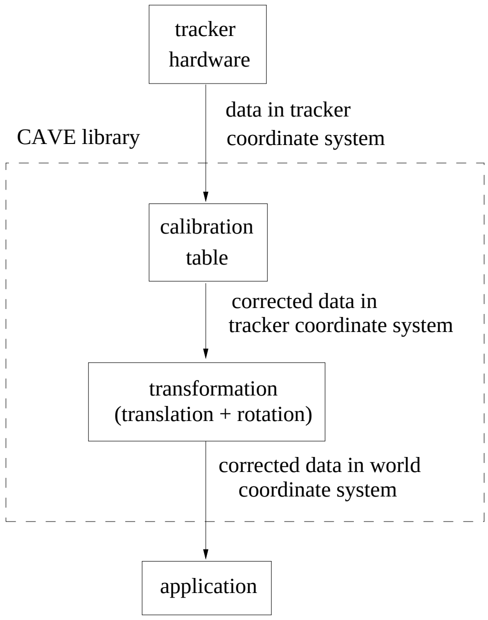

Figure 1: Tracker data transformations in our VR systems

draws it). The most obvious indication of an error in a system is a misalignment between a physical object and the drawing of that object. This problem could be traced to errors in the projection equation data.

Some sources of errors are:

1. the 2D image projected on the screen is not calibrated and not linearized
2. incorrect locations of corners of the projection screen in WCS
    (a) erroneous measurement of the size of the screen
    (b) erroneous measurement of the angle of the screen tilt
    (c) incorrect offset between WCS and screen coordinate system
3. incorrect location of users eyes
    (a) inaccurate tracker reading

[1] CAVE and ImmersaDesk are trademarks of the University of Illinois at Chicago

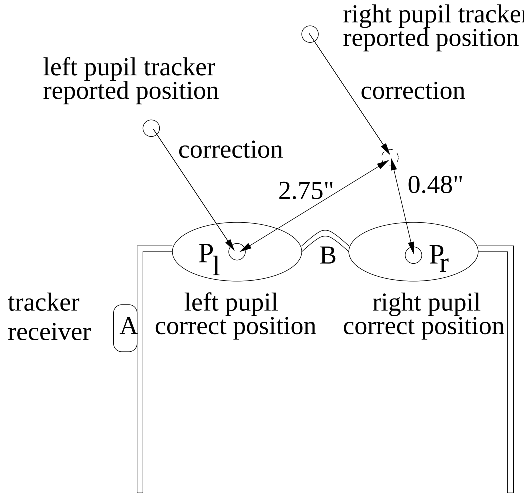

Figure 2: Relation between user's eye positions

(b) incorrect offset between location reported by the tracker and left eye pupil (AP_l Figure 2)

(c) incorrect offset between location reported by the tracker and right eye pupil (AP_r Figure 2)

(d) incorrect transformation between tracker coordinate system and WCS

Some of these errors are insignificant in comparison to tracker errors which are often more than 1 foot.

Current CRT projectors have powerful electronic convergence features. We are able to converge and linearize projectors with a pixel accuracy of 0.1%. The size of the screen can be measured with 1mm accuracy, and the angle of the screen tilt with 1 degree accuracy. These errors total less than 0.1 inch of object displacement for a common viewing situation (user is 3 feet from the ImmersaDesk screen, object is 2 feet in front of the user).

Consider the case of beads suspended above the screen at known locations, with the graphics system drawing their virtual representations. (Figure 3). In a perfect situation the user's eye position and the projection screen location are known precisely. In that case the user will see physical targets aligned with targets drawn (Figure 4) (for simplicity only one eye's relations are drawn and the case is reduced to 2D).

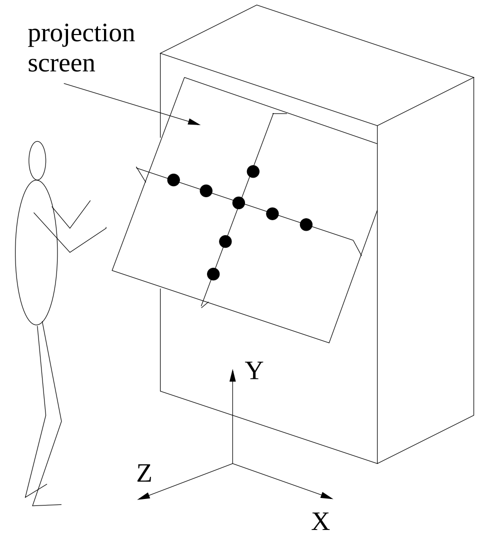

Figure 3: User and ImmersaDesk with physical targets

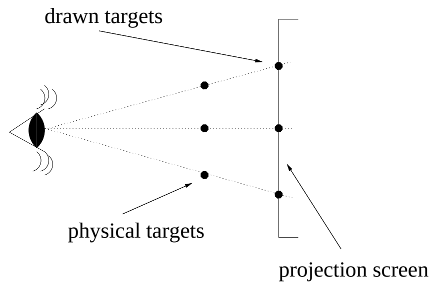

Figure 4: Ideal setup: no errors in location of the screen or user's eye

However in reality there are significant errors in the position of the user's eyes reported by the tracker (Figure 5). Also, the location of the projection screen may not be known precisely (Figure 6). These errors cause misalignment between the physical and virtual targets. The single correction vector for a particular viewing position corrects errors in the lo-

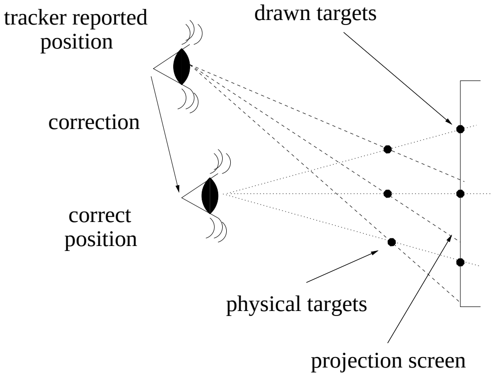

Figure 5: Actual situation: location of user eye is different from the location reported by the tracker

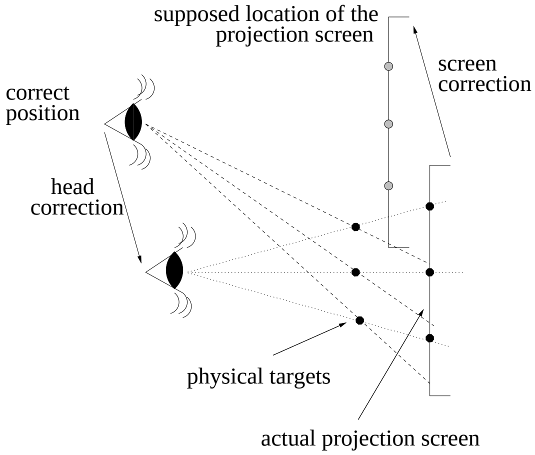

Figure 6: Actual situation: location of the screen is different from previously assumed (Figure 4)

cation of the projection screen and one eye position (items 2a, 2c, 3a, 3b and 3d from the list in paragraph 1.2). In the case of an erroneous location of the projection screen, we are not able to physically move the screen. However, we can make an equivalent correction by moving the reported head position in the opposite direction (Figure 6).

In projection-based systems, the generated image doesn't depend directly on the head orientation (in contrast to HMDs). Only the eye position is important, and it can be corrected with a vector. However, angular tracker errors matter when we start considering both eyes. Usually, there is only one tracker receiver mounted on the glasses, and offsets between the tracker and both eyes' pupils are measured. Therefore it is possible to correct rotational error with a vector correction for one eye, but not for both (Figure 2). This happens because the same correction vector is applied to the left and right eye.

If the user tries to align virtual and real targets using only one eye, for example the left, then misalignment can occur for the other eye (Figure 2). In our setup we notice rotational errors up to 10 degrees, therefore the positional error can be no larger than 0.48 in (assuming an interocular distance of 2.75 in).

Angle information is very important for wand and glove interactions. For example, when the user is holding a virtual object and examining it, angular errors are clearly noticeable. We will address this issue in the future.

Related methods of aligning virtual and real targets have been used for the purpose of calibration of internal parameters in see-through HMDs [Azuma and Bishop, 1994] [Oishi and Tachi, 1994].

2 Method

2.1 Procedure

The procedure starts with calibration and linearization of the projector by placing a transparent sheet with grid lines on the surface of the projection screen. The projector draws the same lines and projector convergence is performed [2]. The offset between the tracker receiver and the pupils are measured (distances BP_R and BP_R on Figure 2). Physical targets are placed in a known locations in front of the screen (Figure 3).

The graphics system draws similar targets in the same locations as the real ones. If there are no errors,

[2]Linearization is not as effective on a monitor, because a standard monitor does not provide extensive convergence and linearization capabilities.

the physical targets will be superimposed on the virtual ones when viewed from any location. However if there is a discrepancy for a particular viewing position the user is able to correct it interactively. The user holds a wand and uses it in a manner similar to a 2D mouse. Using only one eye (left), the x-y plane (left-right, up-down) correction is performed while aligning only the center target (Figure 3). The user keeps his head in the same location and then looks at the side targets and performs z-correction. Because the procedure is performed using only one eye, the user's stereo vision is not used, and interocular distance is not used in the calculation. This process generates the first correction vector $(\mathbf{C}_1 - \mathbf{H}_1)$ for that head position $\mathbf{H}_1$. Based on this vector a uniform lookup table is calculated. For each point $\mathbf{P}$ in the table, a correction vector $f(\mathbf{P})$ is calculated according to the following equations:

$$f(\mathbf{P}) = \sum_{i=1}^{n} \frac{w_i}{\sum_{j=1}^{n} w_j} (\mathbf{C}_i - \mathbf{H}_i) \tag{1}$$

if $dist(\mathbf{H}_i, \mathbf{P}) \neq 0$ for all possible i: $1 \leq i \leq n$;

or

$$f(\mathbf{P}) = \mathbf{C}_i - \mathbf{H}_i \tag{2}$$

if there exists i, for which $dist(\mathbf{H}_i, \mathbf{P}) = 0$

n is the number of corrections (after the first correction n=1), w_i is a weight:

$$w_i = \frac{1}{dist^2(\mathbf{H}_i, \mathbf{P})}$$

and $dist(\mathbf{H}_i, \mathbf{P})$ is the Euclidian distance between $\mathbf{H}_i$ and $\mathbf{P}$:

$$dist(\mathbf{H}_i, \mathbf{P}) = \sqrt{(H_{ix} - P_x)^2 + (H_{iy} - P_y)^2 + (H_{iz} - P_z)^2}$$

The user moves in the environment and makes additional corrections where they seem necessary, in the same manner as the first one. Each time a new correction vector is introduced, the lookup table is recalculated based on equations (1) and (2). Finally, when the user is satisfied, the final table can be saved and used in an application.

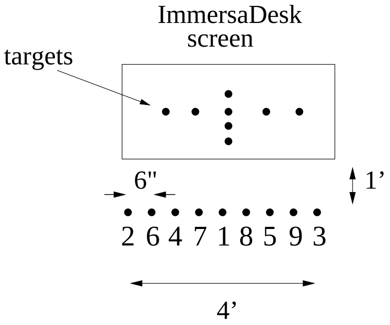

Figure 7: Top view of the experiment site

The above equations imply that a correction vector primarily influences areas close to that vector, and influences more distant areas to a lesser extent. The correction for the point where the user introduced the vector is equal to that vector (equation 2). Based on these equations we can generate a correction table with arbitrary resolution. Usually we use 0.5 or 0.25 feet resolution.

The weight formula was selected after experiments with faster decreasing weights $(\frac{1}{dist^3(\mathbf{H}_i,\mathbf{P})})$ showed that the correction vector did not influence a large enough area, creating a need for too many corrections. A slower decreasing weight $(\frac{1}{dist(\mathbf{H}_i,\mathbf{P})})$ yielded unsatisfactory results as well.

Raw correction vectors $(\mathbf{C}_i - \mathbf{H}_i)$ are saved as well, for possible touch-up calibration later on.

2.2 Results

An experiment was performed on the ImmersaDesk in EVL in order to illustrate how the corrections converge. Vectors were gathered on a line, 6 inches apart at head height (6'), in the order indicated by the numbers in Figure 7. The magnitude of a correction is an indication of the residual error at that head position. Physical targets were constructed with 7 beads suspended 9 inches above the ImmersaDesk

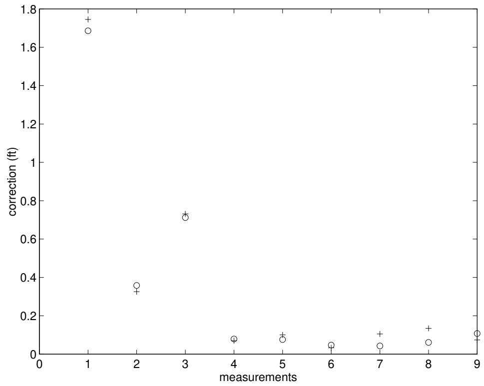

Figure 8: Lengths of correction vectors in the ImmersaDesk environment

screen, laid out in cross shape. The beads were 1 foot apart (Figure 3).

Figure 8 shows the lengths of the correction vectors for each location. Measurements were performed twice (circles and crosses). The first correction is extremely large, because the transformation between the tracker coordinate system and WCS was not estimated accurately. On the right side of the ImmersaDesk there was big metal door frame, which probably caused additional distortions in tracker reading (measurement 3). After just 3 readings the residual error is on the order of 0.1 feet.

A similar procedure was performed in the CAVE. The user was correcting tracker errors at head height (6') in a non-ordered way. The user walked in the environment and made corrections at locations that seemed to have the largest errors. The targets were 3 beads suspended 5 feet above the floor, 2 feet apart, 20 inches from the front wall. Different trials are shown in Figures 9 and 10. It is notable that the user chooses to correct large errors first and continues until satisfactory results are obtained. After 25 readings, the drawn targets seem to be always superimposed on the real ones and corrections are smaller than 0.2 feet. That implies that the errors are smaller than 0.2 feet.

Some of the residual errors are caused by imprecise

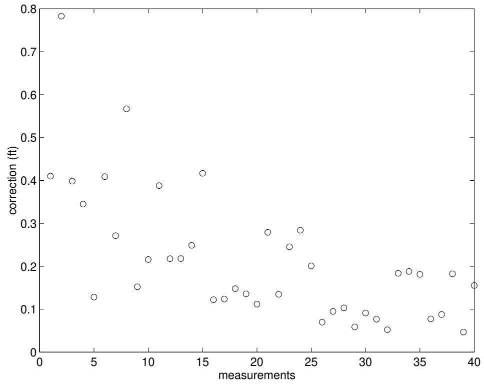

Figure 9: Length of correction vectors in the CAVE

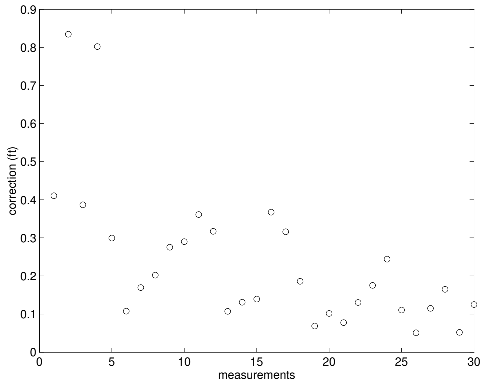

Figure 10: Length of correction vectors in the CAVE

placement of the targets. If the error of placing the target 8 in. in front of the screen is 0.25 in., then viewing it from 3 feet causes an error in eye position of 1.38 in. As mentioned before, rotational errors could cause errors of up to 0.5 in. These errors total 1.87 inches, which is close to the observed errors of 0.2 feet. We are currently working on having smaller and more precisely positioned targets. We also are planning to introduce angular correction to reduce these kind of errors and improve our method.

3 Conclusion

The proposed method allows the user to observe where tracker errors are the largest and perform corrections selectively. Areas that don't have significant errors can be left untouched. This is in contrast to previous methods that required uniform sampling of tracker space. In our method the user can quickly spot the most troublesome areas as well as verify corrections immediately. It is possible to make gross corrections first and add more detailed corrections later. Also, after changes in the environment (for example placing some metal components close to VR environments), some areas could require additional corrections. With this method it is possible to touch up these areas as needed. Only a few 3D locations of the targets have to be measured precisely as compared to 400 with previous methods.

4 Future Work

Angular correction would improve the quality of superimposed drawing of devices like wands or gloves. This problem could be approached in a similar way to position correction. An angular correction table could be build that contains first order rotational corrections. In that case each tracker position in the table would have one angle correction. This part of research is still under implementation.

5 Acknowledgments

This work has been supported in part by the National Science Foundation grant IRI-9424272.

References

[Azuma and Bishop, 1994] Azuma, R. and Bishop, G. (1994). Improving Static and Dynamic Registration in an Optical See-through HMD. In *Proceedings of SIGGRAPH*, pages 197–204.

[Bryson, 1992] Bryson, S. (1992). Measurement and Calibration of Static Error for Three-Dimensional Electromagnetic Trackers. In *SPIE Conference on Stereoscopic Displays and Applications.*

[Cruz-Neira et al., 1992] Cruz-Neira, C., Sandin, D., DeFanti, T., Kenyon, R., and Hart, J. (1992). The CAVE - Audio Visual Experience Automatic Virtual Environment. In *Communications of the ACM 35, 6*, pages 65–72.

[Czernuszenko et al., 1997] Czernuszenko, M., Pape, D., Sandin, D., DeFanti, T., Dawe, G., and Brown, M. (1997). The Immersadesk and Infinity Wall Projection-Based Virtual Reality Displays. In *Computer Graphics 31, 2*, pages 46–49.

[Ghazisaedy et al., 1995] Ghazisaedy, M., Adamczyk, D., Sandin, D., Kenyon, R., and DeFanti, T. (1995). Ultrasonic Calibration of a Magnetic Tracker in a Virtual Reality Space. In *IEEE VR Annual International Symposium (VRAIS).*

[Krueger and Froehlich, 1994] Krueger, W. and Froehlich, B. (1994). The Responsive Workbench. In *Computer Graphics and Applications 14, 3*, pages 12–15.

[Oishi and Tachi, 1994] Oishi, T. and Tachi, S. (1994). Calibration Method of Visual Parameters for See-Through Head-Mounted Display. In *IEEE International Conference on Multisensor Fusion and Integration for Inteligent Systems*, pages 447–454.