Superior Augmented Reality Registration by Integrating Landmark Tracking and Magnetic Tracking

Andrei State, Gentaro Hirota, David T. Chen, William F. Garrett, Mark A. Livingston

Department of Computer Science
University of North Carolina at Chapel Hill

http://www.cs.unc.edu/~us/hybrid.html

ABSTRACT

Accurate registration between real and virtual objects is crucial for augmented reality applications. Existing tracking methods are individually inadequate: magnetic trackers are inaccurate, mechanical trackers are cumbersome, and vision-based trackers are computationally problematic. We present a hybrid tracking method that combines the accuracy of vision-based tracking with the robustness of magnetic tracking without compromising real-time performance or usability. We demonstrate excellent registration in three sample applications.

CR Categories and Subject Descriptors: I.3.7 [Three-Dimensional Graphics and Realism]: *Virtual Reality,* I.3.1: [Hardware Architecture]: *Three-dimensional displays,* I.3.6 [Methodology and Techniques]: *Interaction techniques.*

Additional Keywords and Phrases: Augmented reality, stereo video see-through head-mounted display, frame buffer techniques, registration, calibration.

1 MOTIVATION

While the advent of Head-Mounted Displays (HMDs) and affordable real-time computer graphics engines has given rise to much research in the field of Virtual Reality (VR), comparatively little work has been done in the closely-related field of Augmented Reality (AR). A VR system immerses the user in a totally synthetic, computer-generated environment. An AR system, on the other hand, merges computer-synthesized objects with the user's space in the real world. Synthetic objects enhance the user's interaction with, or his perception of, the real world [Azuma95].

The following are typical requirements for an AR system:

(1) Accurate registration between synthetic and real objects: a virtual object should appear at its proper place in the real world, otherwise the user cannot correctly determine spatial relationships. Dynamic registration is particularly important when the user moves around the environment. The relative position between real and synthetic objects should be constant.

(2) Reasonable image generation rate (10 Hz) and stereopsis: these are important for good depth perception. The lack of kinetic or stereoscopic depth cues greatly reduces the believability of an augmented environment.

(3) Simple initial set up procedure: users of AR applications should not have to be familiar with the specific techniques used in AR systems.

(4) Minimal constraint on user motion: in most applications, the user wants to move without restriction.

(5) Low latency: minimal delay between the user's movement and the display update is required for smooth and effective interaction.

Among these requirements, the accurate registration turns out to be a very difficult problem. Current AR systems cannot convincingly meet this requirement. Typically a virtual object appears to swim about as the user moves, and often does not appear to rest at the same spot when viewed from several different positions.

In current AR systems, most of these registration errors are due to the limitations of the tracking systems [Holloway95]. No conventional tracker satisfies all of the above requirements.

2 PREVIOUS WORK

There has been much research in the field of tracking and registration. Most tracking systems used today in fully immersive VR systems have been magnetic. In the field of computer vision there is a wealth of research on motion tracking.

Today's magnetic trackers are subject to large amounts of error and jitter. An uncalibrated system can exhibit errors of 10 cm or more, particularly in the presence of magnetic field disturbances such as metal and electric equipment. Carefully calibrating a magnetic system can reduce position errors to within 2 cm [Livingston95]. Despite their lack of accuracy, magnetic trackers are popular because they are robust and place minimal constraints on user motion.

Other AR systems have used mechanical [Sutherland68] or optical [Ward92, Azuma94] tracking systems. Both of these systems generally have better accuracy than magnetic trackers, but are burdensome. Mechanical systems tether the user and have a limited working volume, and the optical tracker in [Ward92] requires four dedicated tracking cameras mounted on the user's HMD.

In a video see-through AR system [Azuma95], video images of the user's view are always available. Using those images to track the camera's position and orientation should be

CB #3175 Sitterson Hall, Chapel Hill, NC 27599-3175. Tel: +1-919-962-1700. E-mail: {state, hirota, chen, garrett, livingst}@cs.unc.edu

a reasonable approach, and camera tracking has been extensively investigated in the field of computer vision or photogrammetry. Nevertheless, recovering 3D information from 2D images is not an easy task. An intrinsic problem of computer vision is that an almost infinite number of possibilities must be considered until the images can be interpreted correctly.

Model-based vision assumes a priori knowledge of the 3D geometry of visible objects, reducing the problem from shape recovery to mere camera motion tracking [Lowe87, Lowe92]. Even by simplifying the problem this way, model-based vision methods must still extract object features from images. This typically requires special-purpose image processing hardware to achieve real-time updates. Further acceleration can be achieved through the use of fiducials or landmarks. These artificial "features" of objects simplify image analysis.

The advantage of vision-based tracking when applied to video-see-through AR is that it uses the very same image on which synthetic objects are overlaid. Therefore nearly perfect registration can be achieved under certain conditions [Mellor95, Uenohara95].

The problem of vision-based methods is their instability; to save computation cost, they make numerous assumptions about the working environment and the user's movements, but those assumptions are often impractical. For example, they usually assume temporal coherence of camera movement in order to avoid frequent use of costly search algorithms [Faugeras86, Grimson90] that establish the correspondence between image features and model features. Thus, they usually cannot keep up with quick, abrupt user movements. No vision-based tracker reliably deals with the occlusion of features caused by deformable objects (e.g. hands). Once a vision tracker's assumptions fail, the results can be catastrophic.

Computationally, most vision-based methods use iterative minimization techniques that rely on frame-to-frame coherence. Linearization reduces the problem to a single global solution but requires the vision-based tracker to extract a relatively large amount of information from features or landmarks [Mellor95].

Since image analysis and correspondence finding are costly and error-prone, and because landmarks can be occluded, obscured, or may disappear from the camera's view at any time, it is impractical to attempt to continuously track a large number of features in real time.

3 CONTRIBUTION

We have developed a hybrid tracking scheme which has the registration accuracy of vision-based tracking systems and the robustness of magnetic tracking systems.

We use video tracking of landmarks as the primary method for determining camera position and orientation. This tracking method inherits the accuracy of some vision-based methods, but avoids unnecessary computational cost and reduces the demands on the image analyzer.

Color-coding the landmarks helps the system to quickly identify and distinguish between landmarks. This not only eases system setup and improves performance but also lets the system handle abrupt user movement.

A global non-linear equation solver and a local least square minimizer are used to reduce the burden on the image analyzer. Typically 3 landmarks suffice to determine camera position and orientation. Our formulation gives a universal solution for single and stereo camera cases.

The result of the vision-based tracker is also used for on-the-fly calibration of the magnetic tracker, which assists the rest of the system in four different ways:

Image analysis acceleration: The magnetic tracker helps narrow the landmark search area on images, speeding up the landmark search process.

Selection from multiple solutions: Information from the magnetic tracker is often used to select one of several solutions of a non-linear equation.

Backup tracking: the magnetic tracker acts as the primary tracker if the image analyzer cannot locate enough landmarks. Since the magnetic tracker is *locally calibrated* on-the-fly, we avoid complete loss of registration. If 1 or 2 landmarks (not enough for a unique solution) are detected, several *heuristic methods* are used to minimize registration loss.

Sanity check of the vision-based tracker: As mentioned above, vision-based tracking is sometimes unstable. We avoid catastrophic failure by monitoring the difference between results from the magnetic tracker and the vision-based tracker and discarding corrections that exceed a certain magnitude.

4 SYSTEM HARDWARE

All principal components of our system are commercial, off-the-shelf devices. Our system consists of:

- a Virtual Research VR-4 HMD.
- two Panasonic GP-KS102 CCD video cameras with Cosmicar F1.8 12.5 mm lenses (28° field of view, selected for minimal optical distortion), attached to the HMD.
- an Ascension Flock of Birds™ magnetic tracker with Extended Range Transmitter; the magnetic tracking sensor is attached to the HMD.
- a Silicon Graphics Onyx™ RealityEngine^2™ graphics workstation equipped with a Sirius Video™ real-time video capture device (Sirius), and a Multi-Channel Option™.

The HMD-mounted cameras are 64 mm apart—a typical interpupillary distance for humans—and are oriented with a convergence angle of 4° for sufficient stereo overlap in a tabletop working environment. This angle was chosen for one of our driving applications [State96], which involves manipulation directly in front of the user.

The Sirius captures stereo video images from the head-mounted cameras in real-time and transfers the images to the graphics frame buffer of the RealityEngine^2.

5 SYSTEM OVERVIEW

The hybrid tracker analyzes sensor data from two input streams: real-time video images from the stereo cameras, and tracking reports from the magnetic tracking sensor. The system assumes that the two cameras and the tracking sensor are rigidly interconnected and are rigidly attached to the HMD and the user's head. *Head pose* will refer to the position and orientation of this rigid HMD-cameras-sensor assembly.

We assume that the geometry of this assembly is known and that the transformations between the various coordinate systems (cameras, sensor) have been determined via calibration procedures. We also assume that the world space positions of the landmarks used in the vision-based tracking algorithm are precisely calibrated. All calibration procedures are described in Section 8.

5.1 Operation

For each stereo image pair (i.e. frame), the hybrid tracker attempts to determine the head pose from the landmarks' positions in the images. If this attempt is successful, it determines an error-correcting transformation between the magnetic tracker reading and the head pose computed by the

vision-based tracker. We will refer to this transformation as the *magnetic tracker error.*

The magnetic tracker error computed in one frame is used to predict the head pose in the next frame (temporal coherence). This prediction is subsequently used to compute the expected positions of the landmarks in image space. Figure 1 shows the data flow within the hybrid tracker.

At startup, the magnetic tracker error is initialized to zero. The head pose predictor therefore passes the readings from the magnetic tracker unchanged to the landmark predictor, which computes the expected image-space search areas for the landmarks. Using this data as a starting point, the image analyzer searches for landmarks in the video images.

As soon as the first landmark is detected, the head pose is adjusted via a simple heuristic to line up the detected landmark in image space [Bajura95]. The resulting adjusted head pose—in the case of a single landmark only head orientation is adjusted—is fed back to the landmark predictor for re-prediction of landmark search areas. The system uses these improved values to find additional landmarks, thus iteratively refining its knowledge about the head pose. Each time a new landmark is found, an appropriate head pose adjuster or solver is invoked, depending on the total number of landmarks detected.

There are two distinct cases:

(1) If the number of detected landmarks is not sufficient to completely determine the head pose (under-determined cases), the methods used are local, heuristic position and/or orientation adjusters (Section 7.1) such as the single-landmark method mentioned above.

(2) In well-determined and over-determined cases, a global, analytical solver is invoked (Section 7.2). This solver may compute multiple solutions, in which case a solution selector is invoked. The selector attempts to pick a solution by verifying the consistency of all detected landmarks but is not always able to determine a single best solution. In particular, we often encounter situations in which only 3 different landmarks are visible in both cameras. In such cases we use the sensor reading from the magnetic tracker to determine which solution is correct.

In all cases, under-, well- and over-determined, the computed or adjusted head poses are first subjected to sanity checks. Then they are fed back to the landmark predictor to iteratively detect additional landmarks. This continues until a maximum preset number have been found or until all landmarks in the two stereo images have been found.

The solutions resulting from well- or over-determined cases are stabilized by a local least-square optimizer. If the head pose remains under-determined even after exhaustive search for additional landmarks, the partial correction derived by the most recently invoked heuristic adjuster(s) is retained.

The magnetic tracker error (whether computed and optimized or merely partially corrected) is preserved for head pose prediction in the next frame. This constant, 0^{th} order prediction for the magnetic tracker error is adequate given that our system's frame rates rarely exceed 15 Hz in stereo. We use higher-order prediction (linear, combining the magnetic tracker errors from the 2 most recent frames) only if the application and the tracking environment allow higher frame rates (e.g. non-stereo operation). [Azuma94] showed that higher-order prediction works best at high frame rates.

The corrected head pose delivered by the hybrid tracker yields excellent AR registration between real and virtual objects. Figure 2 shows a view within a video-see-through HMD. A tabletop model with wooden cuboids and landmarks is accurately registered with a computer model of the cuboids (white wireframe lines).

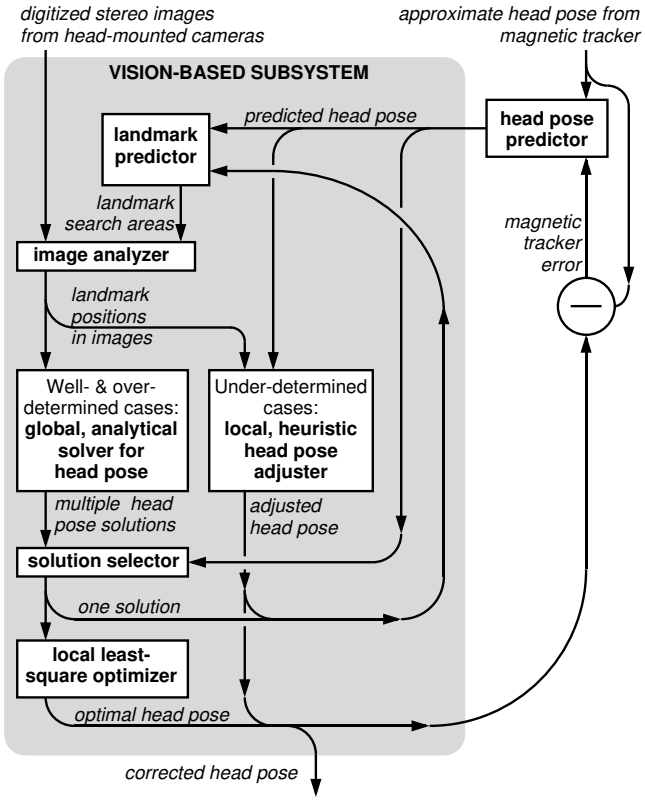

Figure 1. Data flow within the hybrid tracker.

5.2 Vision-only tracking

Vision-only tracking (i.e., without assistance from the magnetic tracker) requires only minor modifications. The predicted head pose delivered to the landmark predictor and to the heuristic adjusters is estimated directly from the head pose in the previous frame(s).

6 LANDMARK TRACKING

The image analyzer detects and tracks landmarks in the video images. Since this is the most time-consuming task in our system, its performance is a primary design concern.

6.1 Landmark shape and color

The landmarks used by the hybrid tracker are two-color concentric circular dots. 11 such landmarks are visible in Figure 2. Each landmark consists of an inner dot and a surrounding outer ring with a diameter that is 3 times larger than the diameter of the inner dot. We use four different colors (mixed from commercial fluorescent fabric paints), which we label as red, green, blue, and yellow; thus we can create 12 unique combinations which can be recognized and identified by the landmark finder.

Color landmarks are useful in several ways. Multiple colors simplify and accelerate low-level pixel inspection, resulting in quick detection. The concentric layout makes our method very robust. While the search algorithm might be easily fooled if it were simply looking for a uniform spot of a single color (as in earlier versions of our system), the more complex structure of two-color landmarks makes spurious detection much more unlikely (Figure 3).

6.2 Operation

The landmark finding subsystem consists of two main components: the landmark predictor, which predicts where the

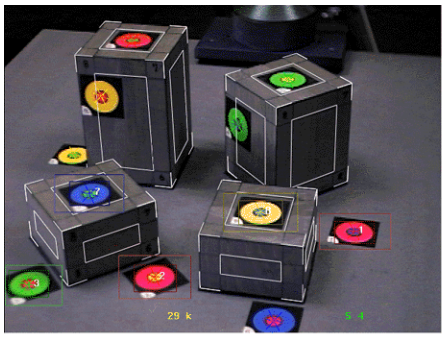

Figure 2. View inside the HMD while the user's head is stationary. The axis-aligned search areas accelerate landmark search. 11 out of the 12 different landmarks created with two-color concentric rings are visible. Note accurately registered computer-generated cuboid outlines (white).

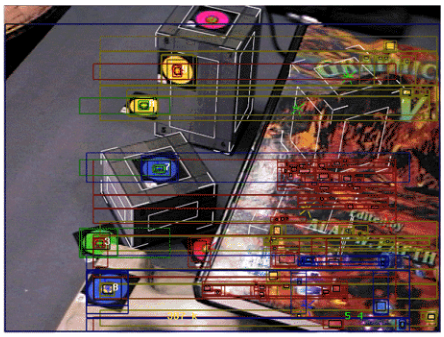

Figure 3. Maintaining registration in the presence of spurious color spots. During landmark search, only specific color and shape signatures are recognized as valid landmarks. Other color areas are inspected but rejected.

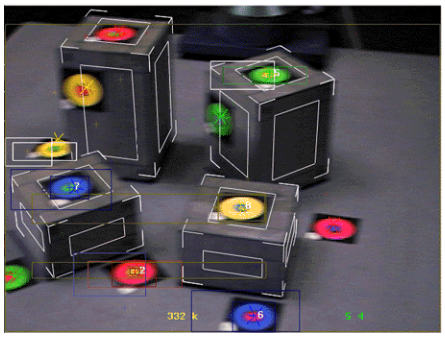

Figure 4. Maintaining registration while the user's head is in motion. Some of the landmarks were not contained within their initial search areas, so the search areas were progressively expanded. Note motion blur.

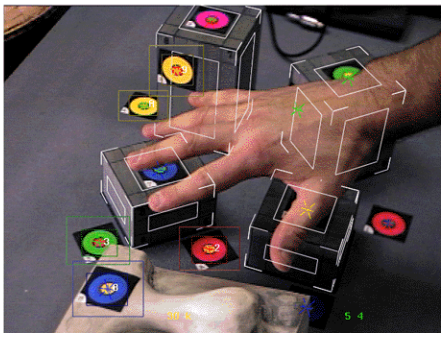

Figure 5. Correct head pose despite landmark occlusion. The landmark tracker is robust enough to handle occlusion. The design of the landmarks makes it possible to detect partial occlusion.

landmarks should be in the video camera image, and the image analyzer, which locates the landmarks in the image.

6.2.1 Landmark predictor

The main task of the landmark predictor is to compute the expected positions and extents of landmarks in image space. For each landmark, a search area is determined based upon the predicted extent. Since the image analyzer operates by exhaustive pixel searches inside search areas, it is important to keep the extents small, i.e., to "tightly" track the landmarks in image space with bounding boxes (Figure 2).

As described above, the hybrid tracker incrementally improves head pose after each newly-found landmark, increasing the accuracy of the predicted positions and predicted extents of the remaining undetected landmarks. As shown in [Bajura95], lining up a single landmark often results in dramatically improved registration. Therefore lining up the first landmark detected often yields accurate search areas for the remaining landmarks, accelerating the subsequent searches. Similar ideas can be found in computer vision literature [Lowe87, Lowe92].

When searching for the first landmark, there are no landmark-derived head pose corrections available, so it is important that the first landmark in each frame be easy to detect. This means the first landmark should have a relatively small search area, and there should be a high probability of actually finding it within that area. To this end, the landmark predictor keeps track of potentially detectable landmarks and sorts them in order of decreasing expected ease of detection. The landmark predictor uses predicted and iteratively improved head poses to compute the expected positions of the landmarks in image space. In addition to this 3D prediction, the landmark predictor performs an internal 2D image space prediction which is not based on input from the magnetic tracker, but only on detected landmarks. For each landmark, the 3D and 2D predictions are compared; if the distance between the two predicted positions is below a preset threshold or if the expected position is far enough from the edge of the image, then the landmark is assigned a high score for ease of detection.

6.2.2 Image analyzer

The second component of the landmark finder is the image analyzer, which starts its search for a landmark by inspecting the search area defined by the landmark predictor.

The first step is pixel marking. Every pixel is classified as belonging to one of the landmark colors or as belonging to no landmark based on the ratios of *RGB* component values. For our specific camera and frame grabber hardware, and under the lighting conditions in our lab, such a simple algorithm can reliably distinguish between only a small number of different colors. We use the four colors mentioned in Section 6.1.

The algorithm looks first for areas whose color matches the color of the outer ring of a concentric landmark and then attempts to locate the inner color dot within the identified area. The marked regions are segmented by horizontal and vertical signature [Haralick93] to determine their centers of mass. If a marked region does not fit inside the bounding box of the search area, the search area is enlarged (Figure 4). For large search areas, a lower sampling density of as little as 1 in 64 (8x8) pixels is used initially; the sampling density is then successively increased as the algorithm reduces the search areas while refining its estimate of the landmark's location.

For all candidate detections consisting of an outer color ring and an inner color dot, two additional tests are performed:

(1) The number of marked pixels in both the inner dot and the outer ring are determined and their ratio is computed. In our case the diameter of the outer ring is 3 times the diameter of the inner dot, so the ratio of marked pixels must be close to 3x3-1=8. If not, the candidate is rejected.

(2) If the centers of mass of the outer and inner regions are not close enough, the landmark may be partially occluded or clipped (explained below). The candidate is rejected.

For accepted candidates, the center of mass of the inner dot is taken as the center of the landmark. Using the center of only the inner dot instead of the average of the centers of the inner and outer areas is advantageous when a landmark becomes partially occluded. In such a case the outer dot will become occluded first, but as long as the landmark passes test (2), the

center will be computed correctly. When the occluding object starts approaching the center dot, the center of mass of the outer ring shifts noticeably, and the candidate fails test (2) and is rejected (Figure 5). If we did not reject these landmarks, then the center would drift before the landmark disappears, corrupting the head pose solutions.

7 HEAD POSE DETERMINATION

Three cases arise when determining the head pose from landmarks. The landmarks represent a set of constraints that is under-determined, well-determined, or over-determined.

7.1 Under-determined case

Until the image analyzer detects at least three different landmarks, the head pose cannot be completely determined from landmarks alone. In these cases, the magnetic tracker is the primary source of information about head pose. A static position calibration lookup table and on-the-fly calibration for the magnetic tracker enable us to use an arsenal of heuristic correctors. These rely on the initial head position being reasonably accurate. After a first rough correction via the predicted magnetic tracker error, a local, heuristic adjustment is applied to the head pose. Different heuristic adjustment methods are used depending on the number of landmarks available.

The heuristic adjusters are designed to ensure highest possible head pose and registration accuracy even when very few landmarks have been detected. They bridge the gap between magnetic-only and vision-based operation of our system. The adjusters are designed to improve head pose as smoothly as possible while more and more landmarks are detected. As a result of this, the hybrid tracker is characterized by reluctant degradation in accuracy when landmarks are lost. When landmarks are re-acquired, the system quickly recovers.

A total of six different under-determined cases exist for our stereoscopic system. The following list describes the basic ideas behind the heuristic adjusters in each case:

(1) Camera 1 sees landmark A, camera 2 sees no landmarks. This is the simple case described and used in [Bajura95]. The method does not adjust head position; it corrects only head orientation by lining up landmark A in the view of camera 1. Only two orientation degrees of freedom can be corrected. The remaining, uncorrected orientation degree of freedom is best described as "rotation about A."

(2) Camera 1 sees two landmarks, A and B, camera 2 sees no landmarks. The method lines up both A and B in the view of camera 1 by reorienting the head. This orientation correction is preceded by a small position correction which is computed to minimize the rotation angle of the following orientation correction. In other words, the head is moved to a position from which the landmarks can be lined up by only minimally changing head orientation. In addition to the slight position adjustment, all three orientation degrees of freedom are corrected.

(3) Camera 1 sees landmark A, camera 2 sees landmark B. This case is similar to (2), except that the two landmarks appear in different camera views. The method lines up A and B in their respective camera views by reorienting the head after the initial position correction. All three orientation degrees of freedom can be corrected. Head position is adjusted slightly, similarly to (2).

(4) Camera 1 sees landmark A, camera 2 sees the same landmark A. The method computes the distance a from the head to landmark A via triangulation in the 2 camera images and adjusts head position by moving the head to the nearest point on a sphere of radius a centered at landmark A. In addition to this position adjustment, two out of the three orientation degrees of freedom can be corrected as in (1).

(5) Camera 1 sees landmarks A and B, camera 2 sees landmark A but not landmark B. This is a hybrid of the methods from (3) and (4). The method triangulates landmark A as in (4), thereby determining its distance a from the head. Then a position adjustment to minimize orientation change is applied as in (3), but with the additional constraint that the position be adjusted towards a point on the sphere of radius a, centered at landmark A's world-space position. In addition to this slight position adjustment, all three orientation degrees of freedom can be corrected as in (3).

(6) Camera 1 sees two landmarks, A and B, camera 2 sees the same two landmarks, A and B. Here the triangulation technique from (4) can be applied to both landmarks, yielding two spheres of diameters a and b, which are centered at their respective landmarks' positions in world space. The two spheres intersect in a circle. The head position is adjusted by translating the head to a point on the circle from which the 2 landmarks can be lined up in the two views by only minimally correcting head orientation. In addition to the slight position change, the three orientation degrees of freedom can be adjusted with a method similar to (2).

The above list shows all possible configurations of 1 or 2 landmarks with a binocular system. As soon as a third landmark is detected in one of the camera views, the system switches to the well-determined case described in the next section.

7.2 Well-determined case

In this section we describe the analytical methods used to determine the head pose when necessary and sufficient information is available from the image analyzer. These methods are based on global equation solvers.

7.2.1 Global solution

Let us consider the head as fixed and the world as attached to landmarks that are moving. The actual head motion can be obtained as an inverse transformation of the landmarks' motions.

We need at least 3 positions of non-collinear points to determine a rigid three-space motion. Therefore 3 non-collinear landmarks are essential. If we find 3 landmarks on the two cameras' image planes, that gives us 3 X-Y coordinate pairs. It is not difficult to see that 6 independent values are sufficient information to determine a 6-degree-of-freedom rigid motion for the head.

Figure 6 shows the geometric relationships between two cameras $\check{C}_1$ and $\check{C}_2$ and three landmarks $\check{L}_1$, $\check{L}_2$ and $\check{L}_3$. The landmarks $\check{L}_1$ and $\check{L}_2$ are detected at $\check{I}_1$ and $\check{I}_2$ in the image of $\check{C}_1$, and the landmark $\check{L}_3$ is detected at $\check{I}_3$ in the image of $\check{C}_2$. The special case in which all three landmarks are detected by one camera can be treated as a case where $\check{C}_1 = \check{C}_2$. Therefore we can consider Figure 6 as the general case.

The unit direction vectors $\check{V}_1$, $\check{V}_2$ and $\check{V}_3$ are obtained simply as: $\check{V}_1 = \frac{\check{I}_1 - \check{C}_1}{\left|\check{I}_1 - \check{C}_1\right|}$, $\check{V}_2 = \frac{\check{I}_2 - \check{C}_1}{\left|\check{I}_2 - \check{C}_1\right|}$ and $\check{V}_3 = \frac{\check{I}_3 - \check{C}_2}{\left|\check{I}_3 - \check{C}_2\right|}$.

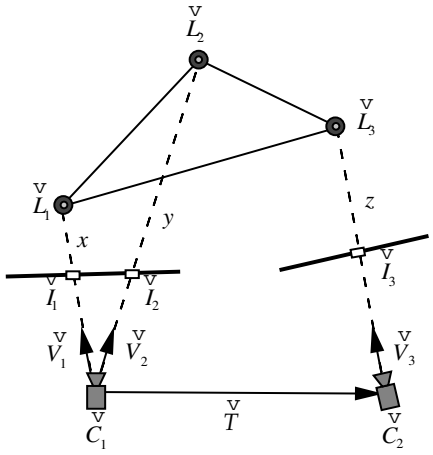

Figure 6. Geometric relationships between three landmarks and the two stereo cameras.

The triangle $\check{L}_1$-$\check{L}_2$-$\check{L}_3$ is undergoing a rigid motion, hence we do not know where it is. But, since we know the positions of $\check{L}_1$, $\check{L}_2$ and $\check{L}_3$ from landmark calibration (Section 8), we can compute the lengths of the 3 edges. They are:

$$L_{12} = \left|\check{L}_2 - \check{L}_1\right|,\ L_{23} = \left|\check{L}_3 - \check{L}_2\right| \text{ and } L_{31} = \left|\check{L}_1 - \check{L}_3\right|.$$

Since both cameras are rigidly mounted on the head set, $\check{T} = \check{C}_2 - \check{C}_1$ is also a constant measured through static calibration (Section 8).

Let x, y and z be $\left|\check{L}_1 - \check{C}_1\right|$, $\left|\check{L}_2 - \check{C}_1\right|$ and $\left|\check{L}_3 - \check{C}_2\right|$ respectively. The result is:

$$\begin{aligned} L_{12} &= \left|x\check{V}_1 - y\check{V}_2\right| \\ L_{23} &= \left|y\check{V}_2 - (\check{T} + z\check{V}_3)\right| \end{aligned} \qquad L_{31} = \left|(\check{T} + z\check{V}_3) - x\check{V}_1\right| \quad (1)$$

Taking the square of both sides of (1) results in:

$$\begin{aligned} a \qquad\qquad &+ b\cdot x\cdot y + x^2 + y^2 = 0 \\ c + d\cdot y + e\cdot z + f\cdot y\cdot z + y^2 + z^2 &= 0 \\ g + h\cdot x + e\cdot z + j\cdot x\cdot z + x^2 + z^2 &= 0 \end{aligned} \quad (2)$$

where aK j are constants given by:

$$\begin{aligned} a &= -L_{12}^2 & d &= -2\check{T}\cdot\check{V}_2 & g &= \left\|\check{T}\right\|^2 - L_{31}^2 \\ b &= -2\check{V}_1\cdot\check{V}_2 & e &= 2\check{T}\cdot\check{V}_3 & h &= -2\check{T}\cdot\check{V}_1 \\ c &= \left\|\check{T}\right\|^2 - L_{23}^2 & f &= -2\check{V}_2\cdot\check{V}_3 & j &= -2\check{V}_1\cdot\check{V}_3 \end{aligned}$$

This is a system of equations consisting of 3 quadratic equations with 3 variables and a total degree of 2x2x2=8. The solutions of this system can be thought of as the intersection of three ellipsoidal cylinders with infinite extents in the x, y and z directions respectively.

If there is only one camera, i.e. $\check{T} = 0$, then d, e and h vanish. In this special case, the following substitution reduces (2) into a system with 2 quadratic equations:

$x' = x/z$ and $y' = y/z$ [Fischler81].

For the general case the solution is more complicated. We use a robust global equation solver that utilizes resultants and polynomial matrices to reduce the system to an eigenvalue problem [Manocha94]. First we eliminate x and y from the system via Dixon's resultant [Dixon08]. The resultant is a determinant of a 6×6 matrix where each element is up to degree 3 in terms of z. The matrix can be written as a matrix polynomial:

$$\mathbf{M}(z) = \mathbf{M}_3 z^3 + \mathbf{M}_2 z^2 + \mathbf{M}_1 z + \mathbf{M}_0 \quad (3)$$

Since $\mathbf{M}_3$ is singular, by substituting $z' = 1/z$ into (3), we get:

$$\mathbf{M}'(z') = \mathbf{M}_0 z'^3 + \mathbf{M}_1 z'^2 + \mathbf{M}_2 z' + \mathbf{M}_3$$

We want z' such that $\det \mathbf{M}'(z') = 0$. We can find solutions for z' as eigenvalues of the companion matrix of $\mathbf{M}'(z')$:

Once we have z', $z = 1/z'$ is plugged into (2), and an (x, y) solution pair that satisfies the three equations can be found.

7.2.2 Selecting one solution

There are eight solutions to our system of equations, so we have to find the most sound one among them. In general, imaginary solutions are trivially rejected, and the physics of the cameras tell us to discard all negative solutions. We typically find two positive solutions. Then the problem is how to disambiguate between these two.

If the image analyzer has detected additional landmarks (that is, in addition to the ones used to solve the equations), we can use these landmarks for disambiguation. Using each remaining candidate solution of the camera, we project the additional landmarks onto the image planes and check how closely the projections match the detected positions. This matching error method works most of the time, but, as shown in [Fischler81], there are degenerate cases in which two or more extra landmarks project to exactly the same position in the image. In addition, errors in landmark detection prevent us from rejecting solutions with small matching errors. However, the most problematic case occurs when we do not have any redundant landmarks, i.e. when we have already used all three available landmarks for equation solving.

In such cases we resort to the aid of the magnetic tracker. Unless the two solutions are very close to each other, we can disambiguate by selecting the solution that best matches the magnetic tracker's readings.

7.3 Over-determined case

Since the equation solver uses only the minimum necessary number of landmarks, it is sensitive to landmark tracking error. Least square error minimization allows us to find an optimum solution using all the detected landmarks. This process neutralizes fluctuations in landmark tracking and significantly stabilizes the final head pose, thereby yielding superior frame-to-frame coherence in registration.

The optimization process is local and depends on the availability of a good initial guess. In any case, the optimizer will converge towards a single solution. It is therefore not advisable to use the optimizer in underdetermined cases, due to the infinite number of solutions. Similarly, in well-determined cases, the number of solutions is finite, but invoking the optimizer would result in convergence towards a single solution. This would preclude inspecting the multiple solutions with the goal of selecting the best one. We therefore invoke the optimizer only when we are confident that a good approximate solution has been found via the methods described in Section 7.2.

The mathematical relationships between the user's head, the head-mounted camera, a landmark and the projected image of the landmark as seen by the camera are:

$$\begin{bmatrix} I_x \\ I_y \end{bmatrix} = \begin{bmatrix} I'_x / I'_z \\ I'_y / I'_z \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} I'_x \\ I'_y \\ I'_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{f} \end{bmatrix} \left[\begin{array}{c|c} \mathbf{R_c} & -\mathrm{R}_c \check{T}_c \end{array}\right] \left[\begin{array}{c|c} \mathbf{R_h} & -\mathbf{R_h} \check{T}_h \\ \hline 0 \;\; 0 \;\; 0 & 1 \end{array}\right] \begin{bmatrix} L_x \\ L_y \\ L_z \\ 1 \end{bmatrix} \qquad (5)$$

In the above equations,

$\check{T}_h$ is a 3D vector representing the position of the head in the world space.

$\mathbf{R_h}$ is a 3x3 rotation matrix representing the orientation of the head in world space.

$\check{T}_c$ is a 3D vector representing the position of the camera in the head coordinate system.

$\mathbf{R_c}$ is a 3x3 rotation matrix representing the orientation of the camera in the head coordinate system.

f is the focal length.

(L_x, L_y, L_z) is the position of a landmark in world space.

(I_x, I_y) is the projected position of the landmark in image space.

(I'_x, I'_y, I'_z) is the projected position of the landmark in homogeneous image space.

I'_x, I'_y and I'_z of (4) can be eliminated using (5). Then (4) can be written simply as

$$F_x = I_x - P_x(L_x, L_y, L_z) = 0$$
$$F_y = I_y - P_y(L_x, L_y, L_z) = 0$$

where P_x and P_y are a combined transformation function that maps a world coordinate to a 2D image coordinate. All values except for $\check{T}_h$ and $\mathbf{R_h}$ are given, therefore F_x and F_y are functions of $\check{T}_h$ and $\mathbf{R_h}$;

$$F_x(\check{T}_h, \mathbf{R_h}) = 0 \text{ and } F_y(\check{T}_h, \mathbf{R_h}) = 0 \qquad (6)$$

Let (t_x, t_y, t_z) be the three components of $\check{T}_h$. $\mathbf{R_h}$ has 9 elements, but a rotation has only 3 real degrees of freedom. This means we can express $\mathbf{R_h}$ as simple rational functions of 3 variables, u, v and w. In our implementation, these parameters are defined as follows. First the initial orientation is converted to a quaternion, then a hyperplane is defined such that it is tangential to the unit hypersphere at the point corresponding to this initial quaternion. Finally u, v and w. are defined as a 3D coordinate system in the hyperplane. Hence (6) can also be written as:

$$F_x(t_x, t_y, t_z, u, v, w) = 0. \text{ and } F_y(t_x, t_y, t_z, u, v, w) = 0 \qquad (7)$$

If we find **n** landmark-projection pairs, using (7) we can set up a system of **2n** equations with 6 variables.

Since I_x and I_y are measured values, F_x and F_y may not vanish. Instead, they should be considered measurement errors in image space.

If the total number of distinct landmarks detected by the two cameras is at least 3, and the total number of landmark-projection pairs detected is at least 4, then this system is overdetermined. In this case we must be able to solve the system as a non-linear, least-square minimization problem using iterative methods. To this end, we incorporated an implementation of the Levenberg-Marquardt algorithm [More80, Fletcher87] into the system. Since a good initial guess is provided by the previously described analytical methods, an optimized solution is computed in only a few milliseconds.

7.4 Non-stereo operation

The hybrid tracker can also operate with a single camera (non-stereo). In that case, none of the binocular solution methods are applied. This means that only heuristic adjusters (1) and (2) from Section 7.1 are used, and only the simplified monocular global three-landmark solver is used. Local optimization is performed using only landmarks visible in one camera.

8 STATIC CALIBRATION

The initial calibration of the system determines numerous static parameters that are required by the tracking procedures described in Sections 5-7. The following list describes the static calibration procedures.

(1) Camera-to-magnetic-sensor transformation: The transformation between a camera and the magnetic tracker's sensor is calculated using an iterative procedure proposed in [Bajura95].

(2) Intrinsic camera parameters: The camera lenses were selected for their low distortion characteristics—well below 1% barrel distortion in the corners of the image. This allows us to keep the mathematical camera model in our system very simple: it is a pin-hole model (no distortion, no skew, 1:1 aspect ratio). This model has only three intrinsic degrees of freedom, which we define as the 3D coordinates of the center of projection with respect to the CCD camera's pixel array. Note that the focal length is in fact equal to one of the three coordinates. We calibrate these coordinates for each camera individually using the vision-based tracker. First we position each camera to see as many landmarks as possible. Then we execute the landmark tracking procedure described in previous sections. The residual error of the least square optimization is an indicator for the accuracy of the estimated intrinsic parameters. An optimization method is then applied to find values for the intrinsic parameters that minimize the residual error. We do not dynamically calibrate the intrinsic camera parameters, because producing reliable results would require tracking considerably more landmarks than our system can identify [Tsai87].

(3) Interocular Transformation: To calculate the transformation between the left and right cameras, we first calibrate the intrinsic parameters as described above. Then we operate the hybrid tracker in dual-mono mode, i.e., by tracking and correcting each camera individually, as described in Section 7.4. In this mode, the transformation between the cameras is not used in the tracking algorithms. It can be computed as the transformation between the cameras' coordinate systems as they are determined by the vision-based tracker. For accurate results, each of the two cameras should see at least three, but preferably more landmarks. We average the data acquired over 10 frames to reduce the effect of landmark tracking errors. This interocular calibration procedure is fast enough for real time execution if desired.

(4) Landmark centers: The world space positions of all the landmark centers are acquired using a precise mechanical arm (FARO Metrecom IND-1).

The FARO mechanical arm is an auxiliary tracker in our system. It is also used to acquire accurate models for real-world objects (for example, the computer model of the cuboids in Figures 2-5). The coordinate system of the mechanical arm must be calibrated to the coordinate system of the magnetic system. To this end, we measure a reference system with both trackers. The reference is a lab-mounted wooden box.

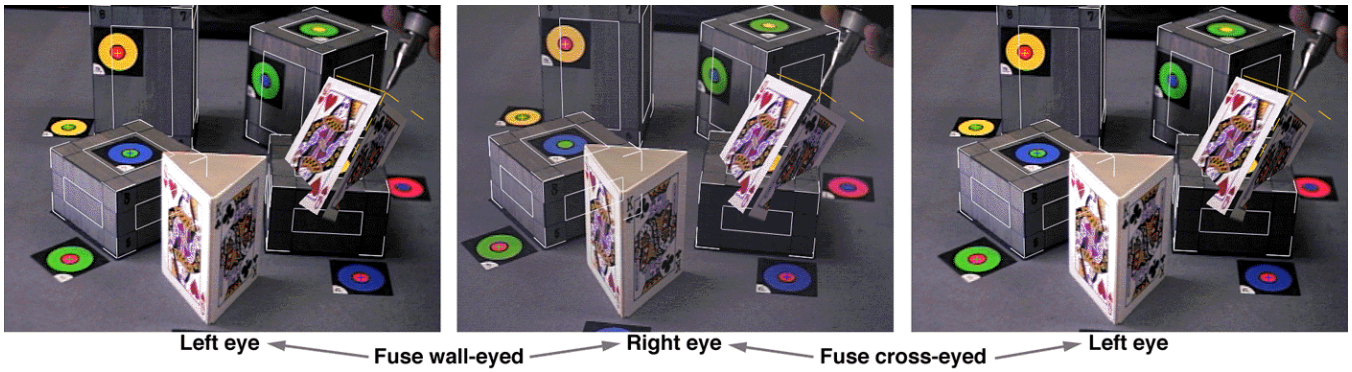

Figure 7. Virtual and real card prisms. Accurate registration makes it possible to acquire an object's texture by projecting the video image onto a precisely registered polygonal model of the object. Notice accurate interpenetration of the virtual card prism and the (real) gray cuboids. The computer-generated white outlines on the cuboids in the background also illustrate the precise registration. Note 3D coordinate axes at the tip of the mechanical arm (top right) used to move the virtual card prism.

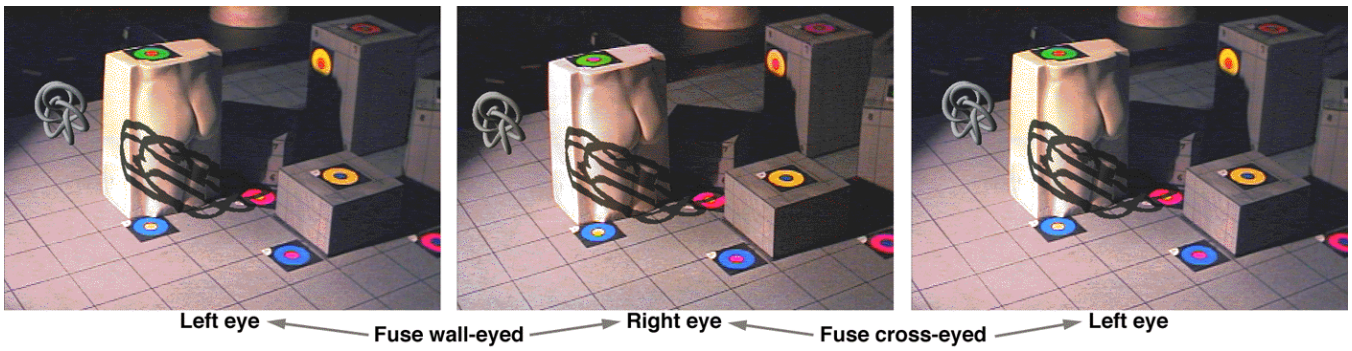

Figure 8. Virtual shadow into the real environment. A polygonal model of the sculpture is registered to the real sculpture. The virtual knot floating beside it casts a (virtual) shadow onto the sculpture and the ground plane. A tracked light source moves real and virtual shadows in sync.

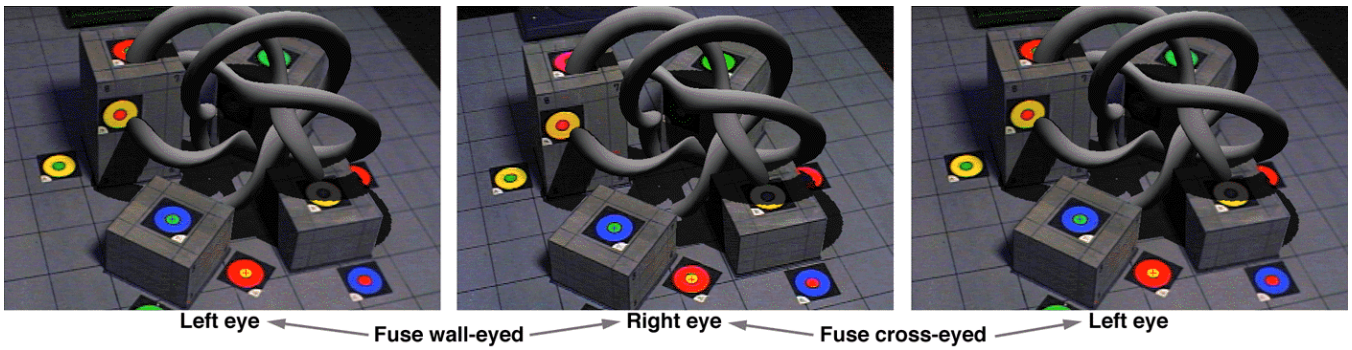

Figure 9. Another example of accurate interpenetration: the virtual knot penetrates into the gray cuboids and also casts virtual shadows into the scene. The landmarks that are occluded by the virtual knot are still used for tracking.

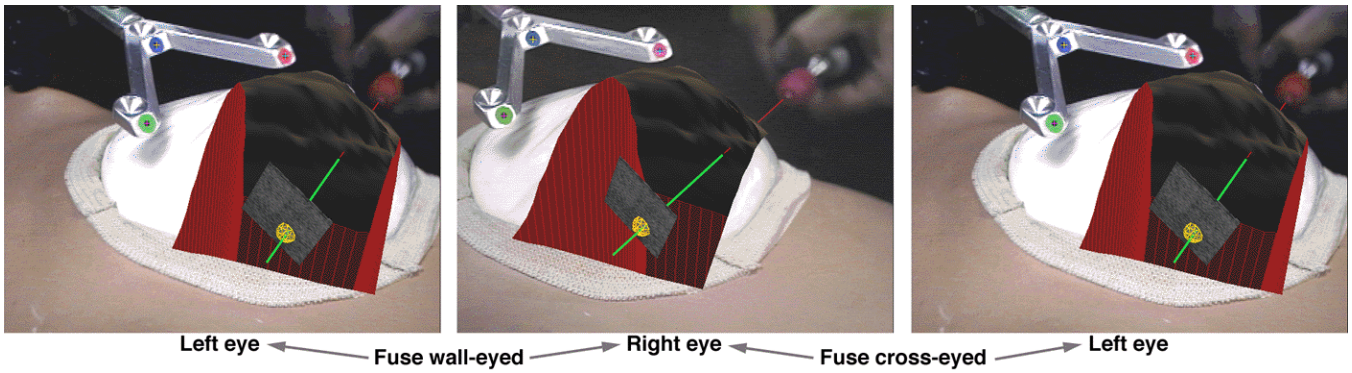

Figure 10. HMD stereo images from an experimental AR system for ultrasound-guided needle biopsy of the breast. A mechanically tracked needle (marked by superimposed red line) is inserted into a training phantom of a human breast. An ultrasound image positioned within the breast is enhanced with a yellow sphere which marks the targeted lesion. Precise stereoscopic registration between the real needle and its virtual extension (green), used for aiming, is essential. The Y-shaped object at the top holds three color landmarks used by an older version of the hybrid tracker.

9 RESULTS AND DISCUSSION

To evaluate the registration performance of our system, we built the tabletop scene shown in Figures 2-5 and 7-9. We register the real world cuboids to computer models. The registration errors are represented by the distances between computer-generated edges and the corresponding real world edges. Typically these errors are below 1 pixel.

We demonstrate the registration accuracy of our method in three experimental AR systems. Figure 7 demonstrates a 3D copy and paste operation in which a virtual copy is made of a real object. The user manipulates the virtual copy of the card prism. Notice that the virtual prism intersects with the real cuboids in a convincing manner. For the 3D copy operation, the real card prism is measured with the mechanical arm. After each face is defined by digitizing its vertices, a texture is extracted from the camera image and is applied to the face.

Figure 8 demonstrates a virtual object, a knot, casting a shadow on a real object, a sculpture. The geometry of the sculpture was digitized with the mechanical arm and placed in the scene. The (real) light source is tracked (by the mechanical arm), and the shadow map is calculated in real-time [Segal92]. Figure 9 shows a similar scene. The knot intersects the real objects, emphasizing the accurate registration of the synthetic imagery (the knot and its shadow) with the real cuboids.

We have also used hybrid tracking in an experimental system designed ultimately to aid physicians in performing ultrasound-guided needle biopsies [State96]. In such a procedure the physician may be attempting to pierce a suspicious lump in a patient's breast. Traditionally ultrasound echography images are used to locate the lump and aim the needle. Our experimental system creates a virtual display of the lump and the needle. The ultrasound image slice which also contains a computer-enhanced display of the target lump, appears to lie within the patient, correctly registered in space. Figure 10 shows images from this system. A mechanically tracked needle is being inserted into a training phantom of a human breast. This system uses an early version of our hybrid tracker that did not use two-color concentric landmarks.

It is difficult to quantitatively determine the final camera position and orientation error in an AR system. It is nearly impossible to evaluate the accuracy of the intrinsic camera parameter calibration and of the interocular transformation calibration procedures. This is due to the fact that ground truth values are unavailable. We have therefore implemented a simulator for the camera video images. The simulator generates synthetic stereo images, complete with landmarks. The intrinsic parameters for the (simulated) cameras are user-settable, as are landmark calibration errors, landmark tracking errors, and magnetic tracker errors. Using the simulator, we determined that the intrinsic parameter calibration is very sensitive to landmark tracking errors and landmark calibration errors. We also determined that intrinsic parameter errors affect interocular calibration accuracy.

The system's final camera position and orientation errors when used in the tabletop cuboids environment are generally below 2 mm and 0.2 degrees (simulator data). This assumes very accurate landmark calibration and image analysis. In practice, camera pose errors are larger but seldom exceed 1 cm and 1 degree in overdetermined cases. It is important to note that in this system—as opposed to AR systems in general [Holloway95]—the effects of position and orientation errors are not cumulative. Instead, they neutralize each other's influence on registration accuracy in the region of space containing landmarks. It follows that our system's registration accuracy is in large part due to the design decision to track landmarks in the target images.

10 FUTURE WORK

Our system is not without limitations. The most important of these is suboptimal performance due to the lack of synchronization between the magnetic tracker and the vision-based subsystem. The magnetic tracker's readings lag behind the camera video images, which makes the magnetic tracker error grow beyond reasonable values if the head moves quickly. Since the landmark predictor does not compute useful landmark search areas in such cases, this leads to full-screen searches and thus to noticeable glitches. The obvious way to reduce the influence of lag is by using a faster head tracker [Mine93a] and sophisticated prediction algorithms [Azuma94]. Delaying the video images [Bajura95] is also possible but undesirable since it increases overall system latency.

Additional though less severe synchronization problems are due to sequential scanout in the video cameras [Mine93b]. Our system does not account for the 17-msec time difference between the top and the bottom scanlines of the video images. Nor does it compensate for the latency difference between the left and right camera video images The effects of such latency differences could be reduced by time-stamping detected landmarks and by reformulating the head pose correctors and solvers to exploit the time stamps.

Under even lighting conditions (Figure 7), the image analyzer can easily recognize our fluorescent landmarks. But despite the use of adaptive brightness evaluation for each landmark, harsh or changing lighting conditions (Figure 8) noticeably diminish the analyzer's performance. Landmark recognition reliability and tracking accuracy could be improved by building constant-intensity landmarks, such as active (for example back-lit) fiducials, or by using retro-reflective materials in combination with an HMD-mounted light source.

A more realistic camera model incorporating optical distortion should make the system usable with wide-angle lenses, thus providing a wide field of view and large stereo overlap. To determine the image-space landmark centers more accurately in wide-angle views, perspective correction should be performed on the centers' coordinates.

Finally, our wish list also includes: attaching landmarks to moving objects (in order to track object motion simultaneously with camera position and orientation), using the system at a different scale (for example, in a room-sized environment), and real-time tracking of visually unobtrusive natural features.

ACKNOWLEDGMENTS

We wish to express our gratitude to Ronald T. Azuma, Michael Bajura, David C. Banks, Gary Bishop, Stephen and Clara Chen, D'nardo Colucci, Henry Fuchs, Arthur Gregory, Stefan Gottschalk, David Harrison, Marco Jacobs, Fred Jordan, Kurtis Keller, Amy Kreiling, Shankar Krishnan, Alan Liu, Dinesh Manocha, Mark McCarthy, Michael North, Stephen M. Pizer, Scott Pritchett, Russell M. Taylor II, Bruce Scher, Chris Tector, John Thomas, Greg Turk, Peggy Wetzel, Mary C. Whitton, Scott Williams, Steve Work, and Silicon Graphics, Inc.

We thank the anonymous reviewers for their comments and criticism.

This work was supported in part by ARPA DABT63-93-C-0048 ("Enabling Technologies and Application Demonstrations for Synthetic Environments"). Approved by ARPA for Public Release—Distribution Unlimited. Additional partial support was provided by the National Science Foundation Science and Technology Center for Computer Graphics and Scientific Visualization (NSF prime contract 8920219).

REFERENCES

Azuma, R. A Survey of Augmented Reality. *SIGGRAPH 1995 Course Notes #9* (Developing Advanced Virtual Reality Applications).

AZUMA, R., BISHOP, G. Improved Static and Dynamic Registration in an Optical See-through HMD. Proceedings of SIGGRAPH 94 (Orlando, FL, July 24-29, 1994). In *Computer Graphics* Proceedings, Annual Conference Series, 1994, ACM SIGGRAPH, pp. 197-203.

BAJURA, M., NEUMANN, U. Dynamic Registration Correction in Video-Based Augmented Reality Systems. *IEEE Computer Graphics and Applications* (September 1995), pp. 52-60.

DIXON, A.L. The Elimination of Three Quantics in Two Independent Variables. *Proceedings of the London Mathematical Society*, 6 (1908), 49-69, pp. 209-236.

DRASCIC, D. ARGOS: A Display System for Augmenting Reality. *ACM SIGGRAPH Technical Video Review, Volume 88: InterCHI 1993 Conference on Human Factors in Computing Systems* (1993).

FAUGERAS, O.D., HEBERT, M. The Representation, Recognition and Locating of 3-D Objects. *Int. J. Robotics Res.,* 5:3 (1986), pp. 27-52.

FISCHLER, M.A., BOLLES, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24:6 (1981), pp. 381-395.

FLETCHER, R. *Practical Methods of Optimization.* John Wiley and Sons, Inc., New York (1987).

GRIMSON, W.E.L. *Object Recognition by Computer: The Role of Geometric Constraints.* MIT Press, Cambridge (1990).

HARALICK, R.M., SHAPIRO, L.G. *Computer and Robot Vision, Volume I*, Addison-Wesley (1993), p. 48.

HOLLOWAY, R. *Registration Errors in Augmented Reality Systems.* Ph.D. dissertation, University of North Carolina at Chapel Hill (1995).

JANIN, A., ZIKAN, K., MIZELL, D., BANNER, M., SOWIZRAL, H. A videometric tracker for augmented reality applications. *Proceedings of SPIE,* November 1994 (Boston).

KANCHERLA, A.R., ROLLAND, J.P., WRIGHT, D.L., BURDEA, G. A Novel Virtual Reality Tool for Teaching Dynamic 3D Anatomy. *Proceedings of CVRMed '95* (Nice, France, April 3-5, 1995) pp. 163-169.

LIVINGSTON, M., STATE, A. Improved Registration for Augmented Reality Systems via Magnetic Tracker Calibration. *University of North Carolina at Chapel Hill Technical Report TR95-037* (1995).

LOWE, D.G. Three-Dimensional Object Recognition from Single Two-Dimensional Images. *Artificial Intelligence*, 31 (1987), pp. 355-395.

LOWE, D.G. Robust Model-based Motion Tracking Through the Integration of Search and Estimation. *International Journal of Computer Vision*, 8:2 (1992), pp. 113-122.

MANOCHA, D. Solving Systems of Polynomial Equations. *IEEE Computer Graphics and Applications* (March 1994), pp. 46-55.

MELLOR, J.P. Realtime Camera Calibration for Enhanced Reality Visualization. *Proceedings of CVRMed '95* (Nice, France April 3-5, 1995), pp. 471-475.

MINE, M.R. Characterization of End-to-End Delays in Head-Mounted Display Systems. *University of North Carolina at Chapel Hill Technical Report TR93-001* (1993a).

MINE, M.R., BISHOP, G. Just-In-Time Pixels. *University of North Carolina at Chapel Hill Technical Report TR93-005* (1993b).

MORE, J.J., GARBOW, B.S. HILLSTROM, K.E. User Guide for MINPACK-1. *Argonne National Laboratory Report ANL-80-74* (1980).

SEGAL, M., KOROBKIN, C., VAN WIDENFELT, R., FORAN, J., HAEBERLI, P. Fast Shadows and Lighting Effects Using Texture Mapping. Proceedings of SIGGRAPH '92 (Chicago, IL, July 26-31, 1992). In *Computer Graphics*, 26, 2 (July 1992), ACM SIGGRAPH, New York, 1992, pp. 249-252.

SILICON GRAPHICS, INC. *Sirius Video Technical Report.* Silicon Graphics, Inc., Mountain View, CA (1994).

STATE, A., LIVINGSTON, M., GARRETT, W.F., HIROTA, G., WHITTON, M.C., PISANO, E.D.(MD), FUCHS, H.. Technologies for Augmented-Reality Systems: Realizing Ultrasound-Guided Needle Biopsies. Proceedings of SIGGRAPH '96 (New Orleans, LA, August 4-9, 1996). In *Computer Graphics* Proceedings, Annual Conference Series, 1996, ACM SIGGRAPH.

SUTHERLAND, I.E. A Head-Mounted Three Dimensional Display. *Fall Joint Computer Conference* (1968), pp. 757-764.

TSAI, R. Y. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. *IEEE Journal of Robotics and Automation*, RA-3:4 (August 1987), pp. 323-344.

TUCERYAN, M., GREER, D.S., WHITAKER, R.T., BREEN, D.E., CRAMPTON, C., ROSE, E., AHLERS, K.H. Calibration Requirements and Procedures for a Monitor-Based Augmented Reality System. *IEEE Transactions on Visualizations and Computer Graphics,* 1:3 (September 1995), pp. 255-273.

UENOHARA, M., KANADE, T. Vision-Based Object Registration for Real-Time Image Overlay. 1995 Conference on Computer Vision, Virtual Reality and Robotics in Medicine (Nice, France, April 1995), pp. 13-22.

WANG, L.-L., TSAI, W.-H. Computing Camera Parameters using Vanishing-Line Information from a Rectangular Parallelepiped. *Machine Vision and Applications,* 3 (1990), pp. 129-141.

WARD, M., AZUMA, R., BENNETT, R., GOTTSCALK, S., FUCHS, H. A Demonstrated Optical Tracker with Scalable Work Area for Head-Mounted Display Systems. *Proceedings of the 1992 Symposium on Interactive 3D Graphics* (Boston, MA, March 1-April 1, 1992), pp. 43-52.

YOO, T.S., OLANO, T.M. Instant Hole (Windows into Reality). *University of North Carolina at Chapel Hill Technical Report TR93-027* (1993).