A Self-Tracking Augmented Reality System

Ulrich Neumann and Youngkwan Cho

Computer Science Department
University of Southern California

uneumann@graphics.usc.edu
ykcho@graphics.usc.edu

Abstract

We present a color-video-based augmented reality (AR) system that is designed to be self-tracking, that is, it requires no separate tracking subsystem. Rather, tracking is performed strictly from the video images acquired through the lens of the camera also used to view the real world. The methods for tracking are rooted in prior research in photogrammetry and computer vision. This approach to tracking for AR systems enables a variety of new applications in assembly guidance that are not feasible with current AR technology. Our initial application is in aircraft manufacturing. We outline our approaches to feature detection, correspondence, pose determination, and system calibration. The results obtained thus far are summarized along with the problems we encountered.

1. Introduction

Augmented reality (AR) systems display virtual objects embedded in a real environment. The AR concept and implementations date back to Ivan Sutherland's "Ultimate Display" [Sutherland 68]. Recent systems illustrate the potential utility of AR visualizations and interactions for medical data display [Bajura, Fuchs, Ohbuchi 92] [Mellor 95] [Wells *et al.* 93], manufacturing [Caudell, Mizell 92], and training applications [Feiner, MacIntyre, Seligmann 93] [Tuceryan *et al* 95]. Arguably, the utility of AR systems for these and other applications is primarily the capability for *programmable accurately-aligned annotation* of real 3D objects. This is in contrast to virtual reality (VR) applications where *simulation and visualization* are often the primary goals. This distinction of ultimate purpose and application utility is what makes AR systems much more dependent on the tracking function than VR systems. Proper alignment of annotation and real objects is a fundamental requirement for AR systems. In many applications misalignment does not just *reduce* the system's utility, it actually conveys incorrect information (with potentially dangerous consequences) – the effect can be worse than no information at all. Therefore, a major requirement of any AR system is to maintain accurate alignment of annotation and real world objects.

Annotation alignment is important, but it can be difficult to achieve within the reality of an application environment. Our goal is providing visual guidance and instruction to human assembly workers in manufacturing environments. Initially, we are focusing on aircraft manufacturing because of the challenges and potential benefits. The application scenario forces us to confront issues such as lighting variations, occlusion, varied scales of operation, and calibration to subassemblies. Figure 1 is an illustration of an application that requires annotation (stay-out instructions) to appear next to holes that are not to be drilled at this stage of assembly. Currently, workers use large blueprints (Fig. 2) that indicate hundreds (or thousands) of holes. Each hole is marked using one of many (10 - 50) possible symbols that summarize when and how the hole is to be created. Workers must interpret the blueprints and locate corresponding positions on the aircraft assembly, which is visible in the upper-right area of figure 2 along with the jig designed to hold it correctly during assembly. This setting is typical of the type of environment in which the tracking problem is difficult to solve.

1.1. Tracking Systems

Commercial magnetic-field-based tracking systems are analog measurement instruments, known to have limited range and accuracy [Meyer, Applewhite, Biocca 92]. Moreover, their susceptibility to interference from EMI and neighboring ferrous material makes them unsuitable for many AR applications in manufacturing or industrial environments.

Arrays of computer controlled LEDs are used in an alternative tracking approach that theoretically is scaleable and offers greater precision [Gottschalk, Hughes 93]. However the system is complex, expensive, and invasive due to the physical infrastructure needed to support the LED arrays.

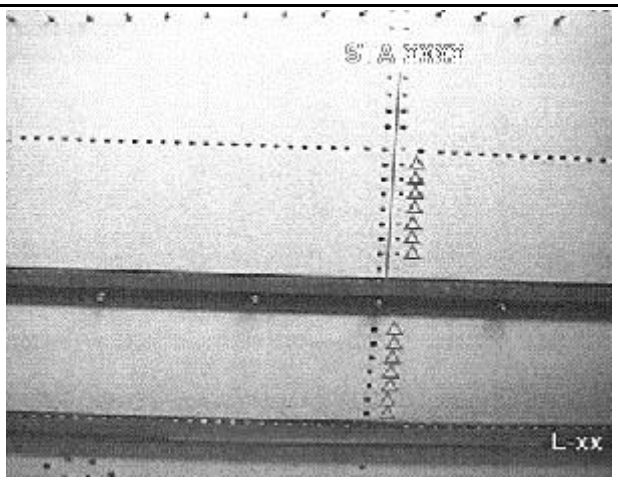

Fig. 1 - Illustration of "stay-out" annotation specifying which holes are not to be drilled at this stage of assembly

Fig. 2 - Assembly workers examining blueprints that specify hole types and locations

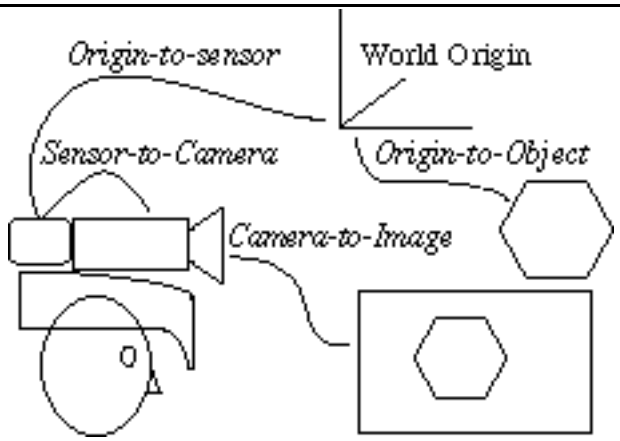

Fig. 3 - Transformations encountered in AR systems with a tracking sensor

The need for motion capture in the entertainment industry has driven the development of systems to track unstructured arrays of active or passive targets from one or more stationary optical sensors [Optotrak 95]. These systems are effective for determining point positions, however orientation must be derived from at least three (non-collinear) fixed-relationship points. Relatively small errors in measuring a point's position can produce unacceptable orientation errors for AR applications. To the best of our knowledge, an assessment of these systems for AR tracking performance has not yet been reported. A recent survey of tracking technologies and issues can be found in [Azuma 95].

To overcome the accuracy, range, and practical problems associated with the tracking systems described above, the system tracks a camera's pose directly from features in the camera's images. The benefits and drawbacks of this approach are detailed in section two of this paper, but its intuitive appeal is that measurements are performed from the perspective of the viewer, not relative to some fixed tracking sensor or emitter. Tracking measurements are made in image space so the measurement error (unavoidable in any tracking method) is minimized in the plane of the image, where the viewer is most sensitive to it, and scaled relative to the size of objects in the image. Another way of stating this observation is that measurement errors are relative to the visually-perceived image space units (pixels), not the world space units (meters, inches, etc.).

Methods exist for determining the pose of a camera based on the positions of N known features in an image. For N=4, unique closed-form solutions exist [Horaud, Conio, Leboulleux 89]. For N=3, up to four solutions exist, frequently collapsing to two. For our initial implementation, we chose the 3-point method of [Fischler, Bolles 81]. Variations of the 3-point pose determination problem date back to 1841 and are surveyed in [Haralick, Lee, Ottenberg, Nolle 94].

1.2. Calibration and AR Display Options

In addition to tracking system inaccuracies, calibration errors also contribute to misalignment. Like tracking, calibration is based on measurements which have limited accuracy. One of the critical calibration measures is the transformation between the tracking sensor's frame of reference and the user's viewing coordinate system (shown as the *Sensor-to-Camera* transformation in figure 3). This can be difficult to measure, especially with optical see-through display systems [Azuma, Bishop 94], where the user is often the only part of the "system" that is presented with the combined view of the real and virtual scenes. Several difficulties are encountered in the optical see-through option: 1) The user is part of the calibration system. 2) A head-mounted display (HMD) is the only display option, and 3) shifting of the display on the head results in loss of calibration. While there are applications that may require optical see-through systems (e.g., surgery), many applications remain practical with video see-through displays. In fact, several simplifications favor video-based systems: 1) The computer system can "see" the same scene the user observes, facilitating a form of feedback which can be used to calibrate or correct the system [Bajura, Neumann 95]. 2) The calibration between sensor and camera disappears when the camera image is also used for tracking measurements. 3) The display/camera combination can be head-mounted, handheld, or tabletop providing a range of cost and performance alternatives.

Additional calibrations are needed for any video-based AR system [Janin, Mizell, Caudell 93]. The virtual camera modeled in a graphics system is a pinhole projection with a specified field of view. The real camera's focal length *f* must be measured [Tsai 87] so that the virtual camera can match it. The real camera image can also exhibit severe lens distortions [Weng, Cohen, Herniou 92] that cause misalignment. Our calibration methods are described in section 2.4.

2. System Operation

A self-tracking system produces aligned annotation by determining the camera pose from features that are both detected and recognized in the camera's image. Therefore, known recognizable markings or fiducials must be visible in the image if annotation is to be added. This may seem like an intolerable constraints until one gets comfortable with the idea that fiducials are just part of the tracking system, like a magnetic tracker sensor and emitter pair, or a ceiling full of LEDs. Once viewed as components of the AR system, many advantages of fiducials over other tracking system components are revealed; fiducials can be small, passive, inexpensive, and liberally placed on and

around the objects we wish to annotate. They are also easily placed on objects moving in the environment. In our assembly application we wish to annotate large non-planar surfaces (ex: 4' x 20') with sufficient precision to identify hole positions to better than 1/4" precision. This translates to roughly one part in a thousand (or 10-bit accuracy) over the entire surface. Our camera has standard video resolution and therefore can not acquire or display images with that precision when the entire surface is visible in one image. However, that precision is easily maintained for partial images of the assembly. (Recall that tracking errors are relative to screen pixels.) During operation, the camera poses range from wide-area views to close-ups of regions-of-interest. Since regions-of-interest often range over the whole assembly, depending on the task, either the distribution of fiducials must be dense enough for worst case viewing, or dynamic placement and calibration of fiducials must be supported. The latter approach is the one we favor since placement of fiducials from the user's standpoint is fast and simple. Many assemblies are also rich in features such as holes, corners, and edges which may be "discovered" by the system and used in lieu of user-placed fiducials.

2.1. Feature Detection

The rows of pilot-holes visible in figure 1 are not there at earlier stages of the assembly process (Fig. 2), so we can not depend on naturally occurring features, and those that are present (*e.g.*, pilot-holes) may not be easily distinguishable from each other. For robustness, we define our own features and a complementary detection process. Currently, our features are solid-color circle and triangle stickers placed in an area of interest. The detection process is model-based, uniquely designed for our features, and summarized below:

1) Subsample the image. Every 8th pixel is examined in a 25×25 pixel window centered on the expected position of a feature, which is currently its position in the last frame. If fewer than three features were located in the previous frame, the entire frame is subsampled .
2) Reject low-intensity pixels as too noisy. The current threshold is 10% of full-intensity.
3) Identify candidate pixels by comparisons against reference color values. References are stored for multiple intensity levels to correct for component-color ratio shifts.
4) Grow the regions around candidate pixels to form candidate regions.
5) Test a region's bounding box for reasonable size (8-100 pixels/side) and aspect ratio (< 5:1).
6) Identify the contact points between a region and its bounding box.
7) Compare the areas of the region and the bounding box to discriminate between circles and candidate triangles.
8) Verify circles by testing for an embedded rectangle formed by bounding-box contact points.
9) Compute a centroid for the feature by weighting the pixel's coordinates by their distance from the reference color. (We consider the centroid position error introduced by perspective projection to be negligible.)

The above method is designed to cull false features as early as possible. The results provide good stability and detection accuracy in fluorescent and outdoor lighting.

The system operates on the notion that the user shares responsibility for robust operation. During the detection process, if feature data seems corrupted or missing, the current digitized image is displayed without annotation, and a new image is acquired for processing. This strategy maintains a high image update-rate when features are missing or obscured. This visual feedback is useful for a user attempting to reposition the camera for proper tracking operation.

As the camera moves, features flow on and off the screen. To make smooth transitions between groups of features, it is important to anticipate the new features so the detection process doesn't have to perform an inefficient full-screen search.

The absolute accuracy of feature positions is difficult to determine. [Mallor 95] gives an error analysis and experimental error measurements for features similar to ours. We measured the repeatability of feature locations for a static camera and scene. Over 40 test were run, each test measured feature positions over ten frames. The maximum variations measured for all tests were ±0.15 pixel in X and Y. The test camera was a SONY CCD FX-230 camcorder, and the digitizer was a SunVideo board in a Sparc20-SX.

Note that we are using full color (24-bit) images. Most of the published work on feature detection methods deal with scalar intensity images. Users, however, want to see the world in color. We believe that it is important to make the transition to color if these systems are to be accepted in real applications and by non-technical people. The transition to color is not trivial and creates a new set of problems.

1) Color component ratios do not track well over varied illumination. Calibration is required.
2) Noise in each color channel is uncorrelated, creating large color-vector variations in low-intensity or shadowed regions of the image.
3) Color transitions may produce leading and trailing color shifts around features. The problem is most apparent with composite-color cameras and is significantly reduced with S-Video and RGB cameras.
4) Pre-emphasis or "detailing" circuits increase the high-frequency gain in many color cameras. This increases the perceived "sharpness" of the image, but it also creates perimeter regions in features with artificially-altered pixel values emphasizing their differences from their neighboring pixels.
5) Computation and data transfer times are increased.

We could use only on the intensity of the color image, but color does not just create problems, it also brings opportunities. Color can be used to more effectively discriminate features from the environment and also from each other, leading to potential simplification of the detection and recognition problem.

2.2. Recognition and Correspondence

Once features are extracted from the image, they must be matched to known feature locations. We currently use a method based on the following information:

1) the feature's location in the previous frame,
2) feature color,
3) triangle and circle discrimination,
3) a list of positions for known features.

The current frame rates (1 - 6 frames-per-second) preclude robust performance while the camera is moving using only the prior-frame feature positions. We expect optimization and more

capable hardware to improve the frame rate and therefore also the robustness of this simple prediction approach. More sophisticated predictions can be computed with state-space filters [Azuma, Bishop 94], and we expect these to become useful once the frame rate reaches 15 frames-per-second or more. (Performance is discussed in section three.)

Color is a primary feature-trait. The system distinguishes three primary (red, green, blue) colors, and three secondary colors (cyan, magenta, yellow). Additional discrimination between circles and triangles allows 12 unique fiducials to be recognized by the current system. By clustering similar subsets of fiducials (*e.g.:* the six circles) with subsets of unique features (*e.g.:* the triangles), many uniquely recognizable clusters (64 in this example) can be created to facilitate unique annotation over a wide region of an assembly.

Although these methods are adequate for our initial tests, both detection and recognition must be more robust before we can consider larger scale assemblies and more varied applications. We are considering the use of concentric two-color features. Other approaches offer possible solution avenues as well. [Uenohara, Kanade 95] primarily make use of the previous frame's feature positions. A constraint is imposed that a group of five features must lie in a plane thereby allowing a geometric area invariant to provide a test for correct correspondences. In its current form this approach also appears to have limited potential for scaling and general application. [Mallor 95] uses the last frame's feature positions as starting points, and if that fails he prunes the search space of a method described in [Huttenlocher 88]. The approach has promise for scalability and generality, but it is coupled to the tracking method and the choice of fiducial marks. Further developments in this area are needed.

2.3. Pose Determination

Pose determination is computed as in [Fischler, Bolles 81]. In practice, we find the method stable over a wide range of viewing conditions. The method involves the solution to a quartic polynomial, so up to four solutions may exist. Discrimination between multiple solutions is performed two ways. If more than three points are visible, the average of the closest set of solutions is chosen. If only three points are visible, proximity to the previous frame's position determines which is chosen. In most three-point cases the number of solutions collapses to two and they are distant from each other. The placement of fiducials is not critical, but unstable poses are known to exist for any triangle geometry, however, the singularities occur along only one viewing direction for each triangle so they rarely occur in practice. The stability of several 3-point pose solution methods is described in [Haralick, Lee, Ottenberg, Nolle 94].

2.4. Calibration

The camera's internal parameters for focal length and lens distortion must be determined. Our method for focal length determination (Fig. 4) uses a planar target with a known grid pattern. Multiple images are taken at measured offsets D along the viewing direction. From each pair of images we compute the focal length *f* using the equations and geometry in figure 4 and average the results. Grid-point pairs closest to the center of the image are used since they are least affected by lens distortion.

$$f = \frac{K_1 K_2 D}{K_1 - K_2} \quad \text{where}$$

$$K_1 = \frac{x_2 - x_1}{d} \quad \text{and} \quad K_2 = \frac{x_4 - x_3}{d}$$

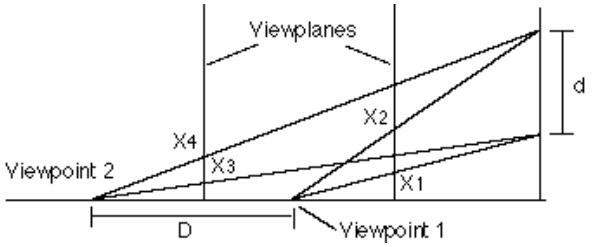

Fig. 4 - Two camera positions suffice for calculating focal length *f*

The distortion calibration method is a variation of the DLTEA-II algorithm [Fan, Yuan 93]. Rather than use an arbitrary camera orientation with 7 independent points, no four of which can be coplanar, we use the planar grid target and perpendicular view pose as described above. The image coordinates (u,v) of a 3D point (x,y,z) can be represented by

$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \lambda \begin{bmatrix} f_x & 0 & U_0 \\ 0 & f_y & V_0 \\ 0 & 0 & 1 \end{bmatrix} [R][I,-T] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (1)$$

where λ is a scale factor relating to perspective projection, f_x and f_y are effective focal lengths along the *u*- and *v*-axes, respectively, and (U_0, V_0) is the center of lens distortion in the image.

R and *T* are components of the view transformation, and due to our simplifications, *R* is identity and all calibration points are in the $z = 0$ plane.

$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \lambda \begin{bmatrix} f_x & 0 & U_0 \\ 0 & f_y & V_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -X_0 \\ I, \ -Y_0 \\ -Z_0 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix} \quad (2)$$

Counting on second order lens distortion, we obtain the following equations

$$U+\Delta U = U + f_x\left(k_1 U_r D^2 + t_1\left(D^2+2U_r^2\right) + 2t_2 U_r V_r\right) = \frac{f_x X_0 + Z_0 U_0 - f_x x}{Z_0} \quad (3)$$

$$V+\Delta V = V + f_y\left(k_1 V_r D^2 + t_2\left(D^2+2V_r^2\right) + 2t_1 U_r V_r\right) = \frac{f_y Y_0 + Z_0 V_0 - f_y y}{Z_0} \quad (4)$$

where k_1 and t_1, t_2 are the coefficients of radial distortion and tangential distortion respectively as described in [Fan, Yuan 93].

$$U_r = \frac{(U - U_0)}{f_x},\ V_r = \frac{(V - V_0)}{f_y},\ D = \sqrt{U_r^2 + V_r^2}, \quad (5)$$

We calculate X_0, Y_0, k_1, t_1, and t_2 by iteratively minimizing the least-square distortion residual.

The remaining calibration issue is to specify how the fiducials and annotations will be calibrated to the actual assembly and each other. At this time, we calibrate the initial known points with a digitizer probe. In the manufacturing setting, jigs that support many assemblies have precision landmarks. We anticipate using those, perhaps with a digitizing probe, to fix an initial set of points on the surface of the assembly. Alternatively, CAD information can supply the positions of features that are designed into the assembly. Much of this task is application and assembly dependent so our approach is to provide a set of flexible options and tools that can be applied as needed in each case.

From multiple views containing at least three known fiducials, we can intersect rays to an uncalibrated newly-positioned fiducial. This allows us to grow the initially-calibrated area or work at multiple scales. For small working areas this method works well, although error propagation, particularly over large areas, remains an issue that must be addressed in our future work.

3. Results and Discussion

A simple test fixture is shown from multiple vantage points in figure 5. The cup is annotated by the square drawn about its rim, about 4.5 inches above the plane of the fiducials. This test simply demonstrates the visual accuracy and tracking volume obtained with the calibration and pose calculations.

The images in figures 6a and 6b show two views of a section of aircraft fuselage. The annotations are the small triangles visible around the windows. The triangles are specified so that one vertex falls in the center of a rivet.

Table 1 lists the time consumed by the video digitizer and our application code, including all the tracking functions for the two resolutions supported by our current system.

Resolution (pixels)	*Digitizer (secs)*	*Application (secs)*	*Frames per sec*
320x240	0.125	0.03	6.4
640x480	0.68	0.26	1.0

Table 1 - Time consumed by video system and AR application

One of the biggest obstacles we foresee for any AR application is the authoring problem. Hand digitizing the locations of fiducials and annotation is tedious. Like CD-ROM or web-page authoring, a specialized environment and set of tools will be needed to facilitate the creation and editing of AR "scripts". Integration and translation of CAD data presents a hurdle. We speculate that AR authoring environments may allow script development by a skilled user's example.

In summary, we developed an AR system that is self-tracking. Our methods for feature detection, correspondence, pose determination, and calibration are described. A challenging application for AR systems is described and initial results are presented. Issues needing further research emerged from this work.

We acknowledge support by NSF grant No. CCR-9502830 and the USC Integrated Media Systems Center, an NSF ERC . We thank Anthony Majoros and McDonnell Douglas Aerospace for their invaluable assistance in defining the applications, for figures 1 and 2, and loan of the DC-10 aircraft section shown in figure 6.

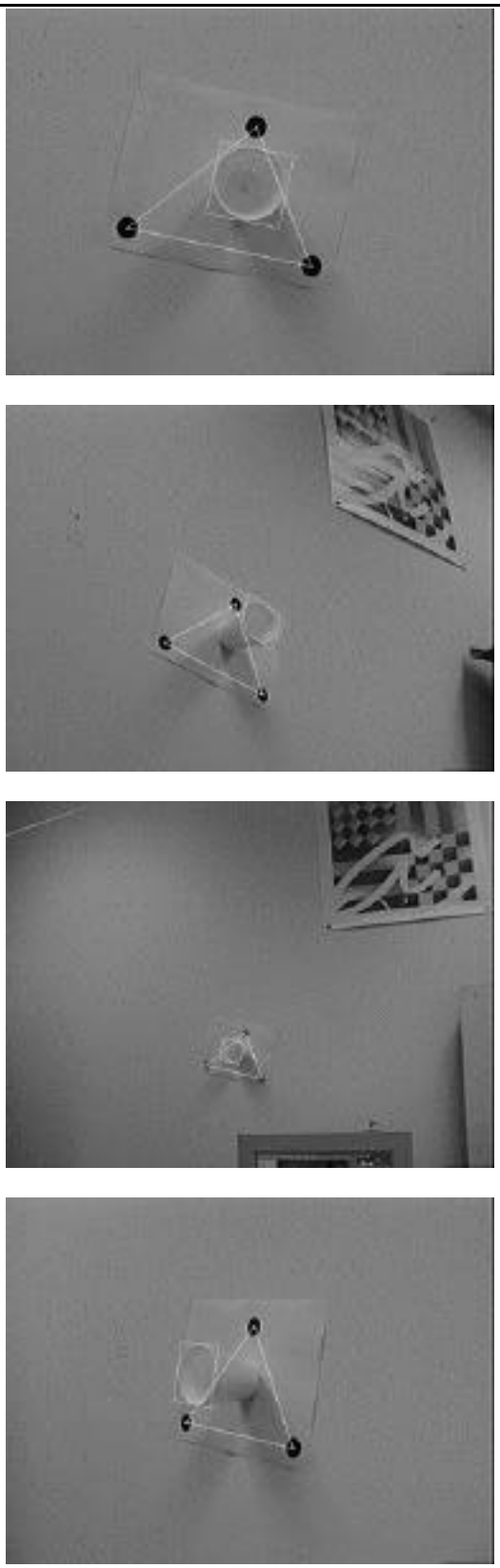

Fig. 5 - Four views of a test fixture convey a sense of the visual accuracy and range of tracking

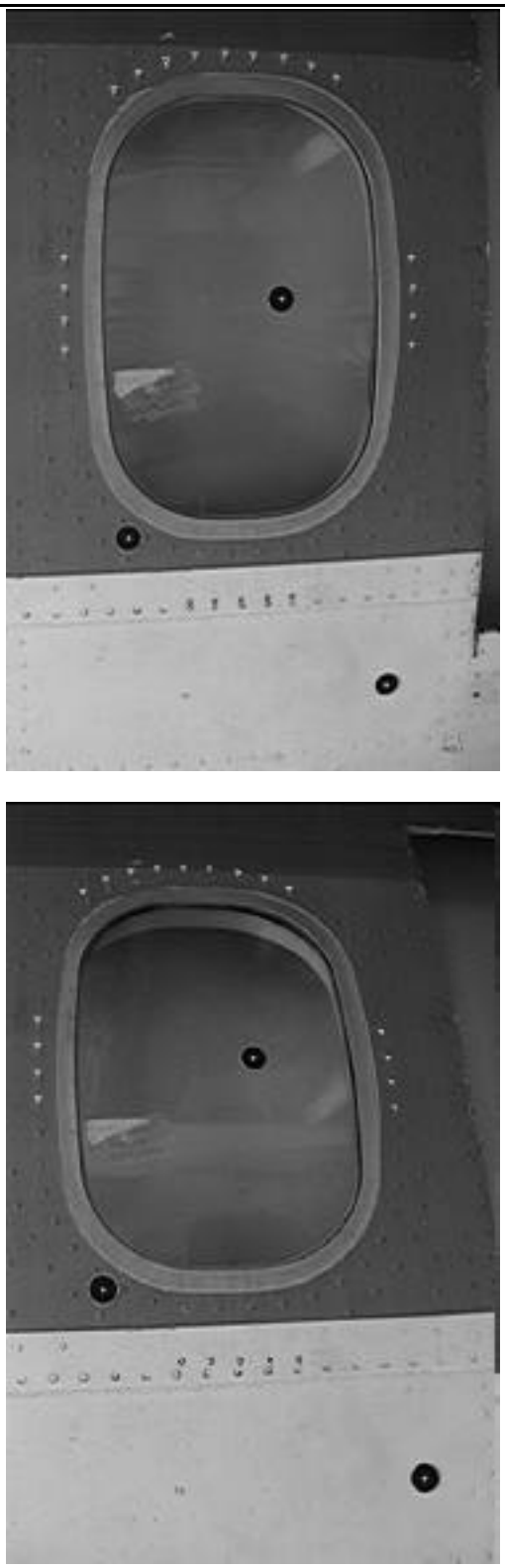

Fig. 6 - Three fiducials and annotation around window frame from two vantage points

4. References

[Azuma 95] Azuma, R. “A Survey of Augmented Reality,” Siggraph 95 course notes, August 1995.

[Azuma, Bishop 94] Azuma, R., Bishop, G. "Improving Static and Dynamic Registration in an Optical See-through HMD," Computer Graphics (Proceedings of Siggraph 1994), pp. 197-204.

[Bajura, Neumann 95] Bajura, M., Neumann, U. “Dynamic Registration Correction in Augmented Reality Systems,” IEEE VRAIS 1995 proceedings, ISBN 0-8186-7084-3, pp. 189-196.

[Bajura, Fuchs, Ohbuchi 92] Bajura, M., Fuchs, H., Ohbuchi, R. "Merging Virtual Reality with the Real World: Seeing Ultrasound Imagery within the Patient," Computer Graphics (Proceedings of Siggraph 1992), pp. 203-210.

[Caudell, Mizell 92] Caudell, T.P., Mizell, D.W. "Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes," Proceedings Hawaii Intl. Conf. on System Sciences, Jan 1992, vol. 2, pp. 659-669.

[Fischler, Bolles 81] Fischler, M.A., Bolles, R.C. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,” Graphics and Image Processing, Vol. 24, No. 6, 1981, pp. 381-395.

[Fan, Yuan 93] Fan, H., Yuan, B. “High Performance Camera Calibration Algorithm,” SPIE Vol. 2067 Videometrics II, pp. 2-13.

[Feiner, MacIntyre, Seligmann 93] Feiner, S., MacIntyre, B., Seligmann, D. "Knowledge-Based Augmented Reality," Communications of the ACM, July 1993, Vol. 30, No. 7, pp. 53-62.

[Gottschalk, Hughes 93] Gottschalk, S., Hughes, J. "Autocalibration for Virtual Environments Tracking Hardware," Computer Graphics (Proceedings of Siggraph 1993), pp. 65-72.

[Haralick, Lee, Ottenberg, Nolle 94] Haralick, R., Lee, C., Ottenberg, K., Nolle, M. “Review and Analysis of Solutions of the Three Point Perspective Pose Estimation Problem,” International Journal of Computer Vision, Vol. 13, No. 3, 1994, pp. 331-356.

[Horaud, Conio, Leboulleux 89] Horaud, R., Conio, B., Leboulleux, O. "An Analytic Solution for the Perspective 4-Point Problem," Computer Vision, Graphics, and Image Processing, Vol. 47, No. 33-34 (1989), pp. 33-43.

[Huttenlocher 88] Huttenlocher, P. “Three Dimensional Recognition of Solid Objects From a Two-Dimensional Image,” Ph.D. thesis, MIT, April 1988.

[Janin, Mizell, Caudell 93] Janin, A.L., Mizell, D.W., Caudell, T.P. "Calibration of Head-Mounted Displays for Augmented Reality Applications," Proceedings of IEEE Virtual Reality Annual International Symposium (VRAIS) 1993, pp. 246-255.

[Mellor 95] Mellor, J.P. “Enhanced Reality Visualization in a Surgical Environment,” Master’s Thesis, MIT A.I Technical Report No. 1544, 1995.

[Meyer, Applewhite, Biocca 92] Meyer, K., Applewhite, H., Biocca, F. “A Survey of Position Trackers,” Presence, Vol. 1. No. 2, Spring 1992 pp. 173-200.

[Optotrak 95] Product literature, Northern Digital Inc. sales@ndigital.com

[Sutherland 68] Sutherland, I. “A Head-Mounted Three-Dimensional Display,” Fall Joint Computer Conference, 1968, pp. 757-775.

[Tsai 87] Tsai, R. “A Versatile Camera Calibration Technique for High Accuracy Three Dimensional 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 *et al* 95] Tuceryan, M., Greer, D., Whitaker, R., Breen, D., Crampton, C., Rose, E., Ahlers, K. “Calibration Requirements and Procedures for a Monitor-Based Augmented Reality System,” IEEE Transactions on Visualization and Computer Graphics, Vol. 1, No. 3, 1995, pp. 255-273.

[Uenohara, Kanade 95] “Uenohara, M. and Kanade, T. “Vision-Based Object Registration for Real-Time Overlay,” Proceedings of CVRMED’95, pp. 14-22.

[Wells *et al*. 93] Wells, W., Kikinis, R., Altobelli, D., Ettinger, G., Lorensen, W., Cline, H., Gleason, P.L., and Jolesz, F. “Video Registration Using Fiducials for Surgical Enhanced Reality,” Engineering in Medicine and Biology, IEEE, 1993.

[Weng, Cohen, Herniou 92] Weng, J., Cohen, P., Herniou, M. "Camera Calibration with Distortion Models and Accuracy Evaluation," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 10, October 1992, pp. 965-980.