Extendible Object-Centric Tracking for Augmented Reality

Ulrich Neumann and Jun Park

Computer Science Department
Integrated Media Systems Center
University of Southern California
{uneumann | junp}@usc.edu

Abstract

This paper presents a novel object-centric tracking architecture for presenting augmented reality media in spatial relationships to objects, regardless of the objects' positions or motions in the world. The advance this system provides over previous object-centric tracking approaches is the ability to sense and integrate new features into its tracking database, thereby extending the tracking region automatically. This lazy evaluation of the structure from motion problem uses images obtained from a single calibrated moving camera and applies recursive filtering to identify and estimate the 3D positions of new features. We evaluate the performance of two filters; a classic Extended Kalman Filter (EKF) and a filter based on a Recursive-Average of Covariances (RAC). Implementation issues and results are discussed in conclusion.

1. Motivation and Introduction

Tracking has been at the heart of research and development in augmented reality (AR) since it's inception in the 1960's [1]. Many systems have been developed to track the six degree of freedom (DOF) pose of an object (or person) relative to a fixed coordinate-frame in the environment [2, 3, 4, 5, 6, 7, 8, 9]. These tracking systems employ a variety of sensing technologies, each with unique strengths and weaknesses, to determine a world-centric pose measurement, as required for virtual and augmented reality applications. Augmented reality, however, differs from virtual reality in that the virtual data or media are often spatially-linked to real objects in the environment. Tracking in a fixed frame of reference, therefore, presents a limitation for augmented realities, since it implies that objects in the environment are calibrated to the tracker's frame of reference and, after calibration, they do not move. This assumption is valid for applications such as architectural visualization [10, 11] where the walls, floors, and doors form a rigid structure whose world coordinates can be calibrated, or are already known by design, and are not likely to move. This rigid-structure criterion excludes a large class of AR applications that provide annotation on objects whose positions in a room or the world may vary freely without impact on the AR media linked to them. For example, AR applications in manufacturing, maintenance, and training [10, 12, 13] require virtual annotations that provide task guidance and specific component indications on subassemblies or portions of structure (Fig.1). These applications are often object-centric, and a more appropriate tracking solution, based on viewing the object itself [14, 15, 16, 17, 18, 19, 20, 21], is provided by the pose estimation methods developed in the fields of computer vision and photogrammetry [22, 23]. A problem with many vision-based tracking methods is that tracking is only possible in a constrained set of views for which certain features of the scene are visible. In this paper we show how to reduce this limitation in many cases, by allowing users to dynamically expand the range of tracked camera views as the AR system is operating.

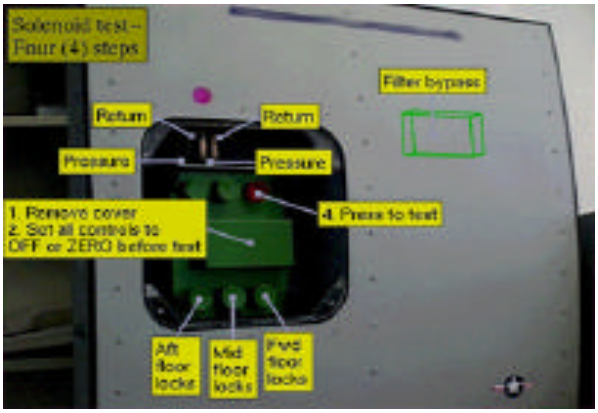


Fig. 1 - Example of AR annotation supporting a maintenance task

World-centric trackers can be used to calibrate movable objects [13, 19, 24, 25], but this generally entails placing and calibrating tracking elements on each object of interest and operating within range of the shared tracking infrastructure, (*e.g.*, magnetic fields or active beacons) [4, 9]. These requirements often make it difficult and expensive to calibrate moving objects with current world-centric tracking approaches.

1.1. Issues in Vision-Based Tracking

Vision-based tracking is the problem of calibrating a camera's pose relative to an object, given one or more images. Many vision-based tracking methods can be coarsely grouped into the three following categories, based on their requirements:

- Three or more known 3D points from a single image [23, 26, 27, 10, 20, 21, 28, 15, 8]
- A sequence of images from a moving camera where point positions may be known or unknown [29, 30, 31, 9]
- Model and template matching from a single image [32 , 14]

The first class of approaches uses a calibrated camera to provide constraints to the pose formulation, so only a few known points within a single image are needed for tracking. This class of methods has been successfully employed in AR tracking systems. Our work and the remainder of this paper are based on these methods as well.

The points needed for tracking can be object features such as corners and holes, or they can be intentionally designed and applied targets or fiducials. We selected the latter option in our work since naturally-occurring features can be difficult to recognize due to their variety and unpredictable characteristics. Also, objects do not always have features where they are needed for tracking; large regions of surfaces are often indistinguishable when viewed without context. Fiducials have the advantage that they can be designed to maximize the AR system's ability to detect and distinguish between them, they can be inexpensive, and they can be placed arbitrarily on objects. The design and detection of fiducials are important topics, but for this work, we require only that fiducials have some basic characteristics such as color and shape.

Our goal is to increase the range of camera-motion that provides tracked viewing while minimizing the number of fiducials that must be applied to an object. A worst-case occurs when all regions of a large object are equally likely to be viewed and therefore a dense distribution of fiducials is needed to support tracking over the entire object. Covering large objects (*e.g.*, airplanes, machinery) with fiducials is impractical and calibrating them is difficult. Our approach is an alternative analogous to "lazy-evaluation" in algorithm design; fiducials are only placed, and their positions computed, as the need for them arises. An initial set of fiducials is strategically placed and calibrated on an object or a fixture rigidly connected to an object. As regions of the object require additional fiducials to support tracking, users simply add new fiducials to those regions of interest and allow the system to automatically calibrate them. Once calibrated, the new fiducials are added to the database of known fiducials and they are used for tracking. This approach is most practical when a limited region of the object needs to be viewed and tracked for a given task, but that region and task is just one of many that may occur. Assembly, training, and maintenance tasks often have this quality. Information for a given task is local, but the specific task and locality is only one of many that may be performed by different people at different times. Figure 1 illustrates AR annotation for a maintenance task that requires tracking in only a limited region around an access panel in a larger structure.

The remainder of this paper presents our vision-based AR tracking architecture. Details are given on the recursive filters that calibrate the 3D positions of new fiducials. We compare the results from two filters and two operational examples.

2. System Architecture

Figure 2 depicts the architecture we are developing. A single camera provides real time video input, and a user observes the camera image and AR media overlay on a display that may be desktop, handheld, or head-mounted. Sections 2.1 - 2.3 describe the components of this architecture that set the context for section 3 where the unique new point evaluation is described.

2.1. Segmentation and Feature Detection

The important aspect of this step is that fiducials must be robustly detected and located. They are defined by a 2D screen position *p* and a type *q* that encodes characteristics such as color and shape. Our fiducial design is a colored

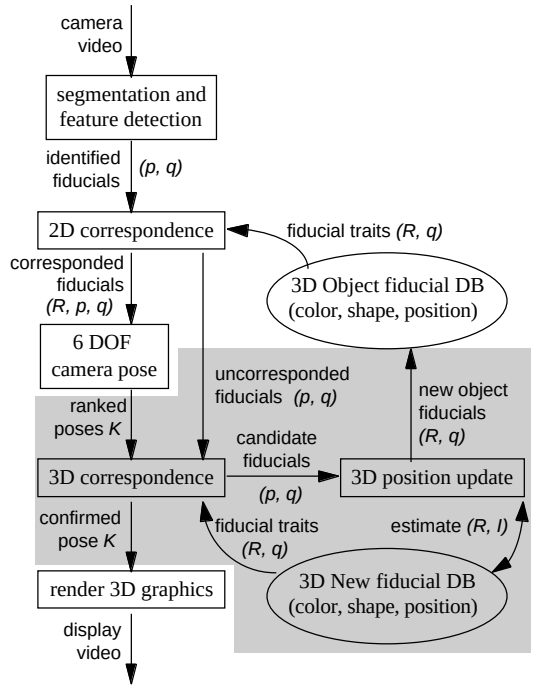


Fig 2 - AR system architecture for extendible vision-based tracking. New fiducial management functions are shown against a gray background.

circle or triangle [15], but other designs such as concentric circles or coded squares are equally valid [7, 33, 10]. We use the three primary and three secondary colors along with the triangle and circle shapes to provide twelve unique fiducial types. Fiducials are detected by segmenting the image into regions of similar intensity and color and testing the regions for the color and geometric properties of fiducials. Detection strategies are often dependent on the characteristics of the fiducials [16, 34, 35].

2.2. 2D Correspondence

The 2D fiducials *(p,q)* must be corresponded to elements in a database of known 3D fiducial positions and types*(R,q)*. The result of a successful match is a corresponded fiducial whose 2D screen position and 3D coordinate on the object are now known *(R,p,q)*. Fiducials observed in the image, but not corresponded to the world database, are passed on as uncorresponded fiducials and potential new points to estimate.

Computing correspondences is hard in the general case [21, 35], and trivial if the fiducial types are unique for each element in the database. For this work, we consider only unique feature types that are trivial to correspond since our focus is on the extendible tracking portion of the architecture, however, we recognize that correspondence is a necessary function for the architecture to scale to truly useful levels. Other researchers have proposed solutions of varying applicability to our case, and we intend to leverage from their work in the future [17, 21].

2.3. Camera Pose

Three or more corresponded fiducials allows the camera pose to be determined [23]. The approach we use has known instabilities in certain poses, and in general provides multiple solutions (two or four) from a fourth-degree polynomial. Methods have been proposed to select the most likely solution [17]. We weight several tests to rank the possible pose solutions in terms of their apparent correctness.

- The distance between the current and last frame's viewpoint is inversely proportional to a solution's weight.
- The fiducials' relative pixel-areas are tested for agreement with the expected relationships based on each pose solution.
- When more than three known fiducials are available, their projections under the candidate poses are compared to their measured positions.
- When more than three known fiducials are available multiple pose calculations are performed for different sets of three fiducials, and the closest pairs from each calculation are highly ranked.

The result of this function is an array of poses *K*, ranked by the above tests, which are passed on to the new point functions (shown over gray in Fig. 2).

3. New Point Evaluation

This section details the unique aspects of this architecture that provide the interactive extension of the tracked viewing range. Given the camera pose and image coordinates of the uncorresponded features, these feature's position estimates are updated in one or two possible ways, as detailed in the next two sections.

3.1. 3D Correspondence

This correspondence function is needed to match uncorresponded fiducials *(p,q)* with current estimates of new fiducials *(R,q)* that are kept in a database. As mentioned in section 2.2, our current system assumes that only one fiducial per color exists. Even though only one feature is used for each color, false features are detected from the noisy environment (e.g., holes in the object may be detected as a circular feature) and these would erroneously-affect new fiducial position estimates if they contribute to the 3D estimate.

In the EKF estimation process, fiducial position updates and error corrections are based on the screen space difference between the fiducial's coordinate in the image and the projection of its corresponding 3D position estimate. Hence, a 2D distance threshold in screen space is able to cull outlyers.

In the RAC estimation process, position updates and error measurements are performed in 3D object space, so an outlyer threshold sphere is determined by the standard deviation of the measurement error that characterizes the system (*e.g.*, 3 SD for 99% acceptance rate).

The initial position estimates from the first few frames must lie within the outlyer boundary of the true fiducial position, or the filters may never converge. We currently require the user to accept a few frames with a button click to ensure valid initial position estimates.

3.2. 3D Position Update

The Extended Kalman Filter (EKF) has been used in many similar applications and the details can be found in references such as [9, 30]. Figure 3a illustrates its operation. The function $\vec{h}$ represents a homogeneous matrix that calculates the projection $\hat{Z}$ of the predicted state $\hat{X}^-$ given the camera pose $\vec{C}$ and the camera intrinsic parameters P_c.

The Recursive Average of Covariances (RAC) filter maintains a position estimate and the covariance of its uncertainty (Fig. 3b). The position update direction and magnitude are obtained from the current estimate, covariance, and a *new line* projected from the new viewpoint through the fiducial's screen position (Fig. 4). The position update direction $\vec{v}$ is computed as

$$\vec{v} = \lambda_1 \ \vec{q}_1 + \lambda_2 \ \vec{q}_2 + \lambda_3 \ \vec{q}_3 \qquad (1)$$

where λ_i are eigenvalues of the current uncertainty covariance and $\vec{q}_i$ are unit vectors from the current position estimate to the closest points s_i along the new line to each eigenvector εi. Figure 4 illustrates s_2, the closest point to $\varepsilon 2$ along the new line.

The current position estimate is updated, by an amount *m*, to the nearest of the uncertainty ellipsoid bound *w* or the distance *d* to the new line along the update vector $\vec{v}$.

$$m = \min(w, d) \qquad (2)$$

Initially the uncertainty is a sphere with radius of 3 SD where SD is the standard deviation of the measurement noise. For each new line, a new uncertainty ellipsoid is modeled and recursively merged with the current uncertainty to update its value.

$$UE = (1-u) \ UE + u \ \ newUE \qquad (3)$$

where *UE* is the uncertainty ellipsoid and *u* is an uncertainty value defined by $u = \lambda_1 \ \lambda_2 \ \lambda_3$ (λ_i are eigenvalues of *UE*). This process gives rise to the name of this method "RAC" (Recursive-Average of Covariances).

When the uncertainty is small enough (e.g., $u < 0.05$), the position estimate of the new feature is deemed usable and the fiducial is added to the database of known fiducials and it can be used for tracking.

3.3 Filter Comparisons and Results

Both filters appear stable in practice. The EKF has known mathematical properties of optimality under certain conditions[30], however the RAC gives comparable results, is simple, and operates completely in 3D world space. A disadvantages of the RAC is that its performance depends on parameter settings to a greater degree than the EKF seems to.

We tested the EKF and RAC filters to find the positions of two new fiducials and, after they are known, we use them to track the camera. We started with three known fiducials and two cases were tested. In the *panning* case,

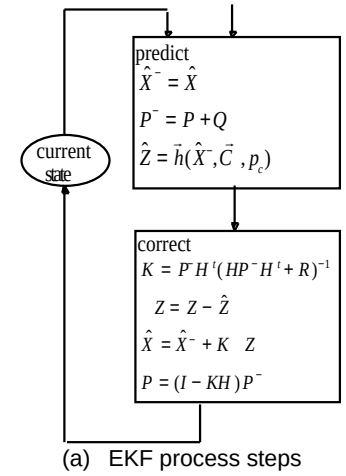


Fig. 3 - A summary of the two recursive filters.
(a) the EKF measures error in screen space
(b) the RAC filter measures error in object space

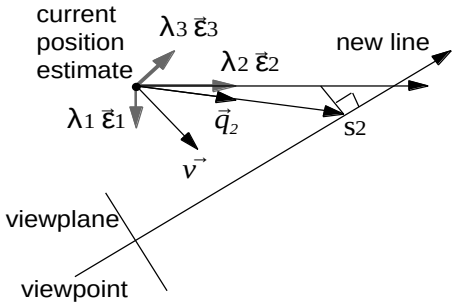


Fig. 4 - The RAC filter updates its position based on the direction of $\vec{q}_i$ vectors oriented towards the points S_i of closest approach between the new line and the eigenvectors

new fiducials were placed to the side of the known fiducials, assuming a translated region of interest (Fig. 5). In the *zooming* case, new features were placed in the center so that the user can zoom in to the region of interest (Fig. 6).

Figure 6 shows images of the zooming case. The initial camera view is shown in (a). Three frames were accepted by the user to converge the EKF to a reasonable estimate (b). The 3D positions of the new fiducials were refined automatically over 42 additional frames. Virtual objects were then rendered using the EKF filter (c) and the RAC filter (d). Figure 5 shows images of a similar sequence of states for the panning case.

Figures 7 and 8 show how the estimates are updated as the frame numbers increase. Blue, green, and cyan spheres represents the positions of known fiducials, and the cone represents the camera position scaled to appear within view. The line from the camera shows the current new line connecting the camera and the new fiducial in the image. The yellow shorter lines represent previous new lines to illustrate the camera's motion. The purple translucent ellipsoid and opaque sphere represent the uncertainty and the current estimate of the EKF while the red ones represents those of the RAC filter. These images demonstrate how the uncertainties shrink and the estimates converge.

Figures 9 and 10 show the distances between the computed 3D position estimates and the digitizer-probe measurements. Both the EKF and RAC results converge to positions offset from the digitized positions. In searching for the source of error, we projected the digitized positions for each camera position and discovered a relatively fixed bias. For example, the red feature in the panning case has a measurement bias of approximately (-4.6, -3.1) screen space pixels. This translates to about 0.24 inches of position offset. We currently suspect the source of this bias to lie in our calibration procedures. Figure 11 shows that when accurate (synthetic) data with noise is supplied to the position estimation process, it converges consistently.

The average distances of the camera from the features were 48.3" for the panning case and 45.9" for the zooming case. Taking account of the focal length (1138 pixel units), the pixel accuracy of the system is about 2.5 pixels.

Our hardware configuration included

- SGI Indy 24-bit graphics system with MIPS4400@200MHz.
- SONY DXC-151A color video camera with 640x480 resolution, 31.4 degree horizontal and 24.3 degree vertical field of view(FOV), S-video output.
- MicroScribe-3D mechanical digitizer with 0.017" accuracy (mean value).

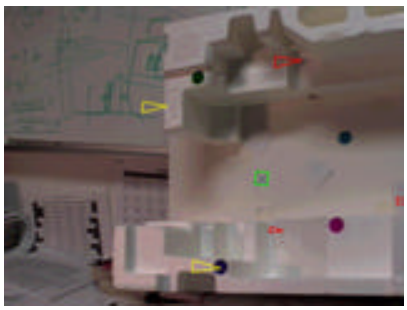
Although the absolute estimates have error, we are encouraged to observe that the estimate variations settle to a steady state without oscillations. This implies that with better camera calibration, greater absolute accuracy can be obtained.

Our future work entails a solution to the 2D correspondence problem. This is necessary for the system to scale to greater numbers of fiducials. We also hope to improve our ability to select the correct pose solution, and we will have to address the error propagation from multiple fiducial estimations.

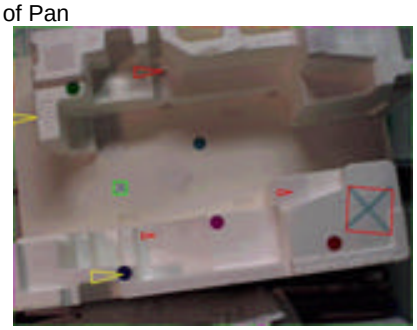
4.0 Acknowledgments

We gratefully acknowledge support funding from NSF grant CCR-9502830 and the Integrated Media Systems Center, a National Science Foundation Engineering Research Center with additional support from the Annenberg Center for Communication at the University of Southern California and the California Trade and Commerce Agency. Thanks also go to Youngkwan Cho, Suya You, Anthony Majoros (The Boeing Co.), and Michael Bajura (Interval Corp.) for technical input, and to the reviewers for their suggestions.

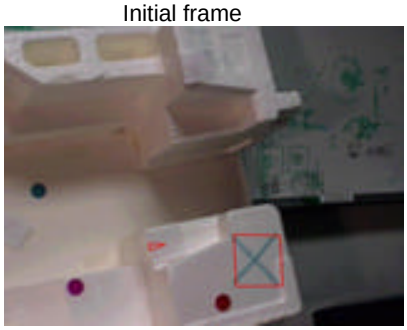
Result of Pan



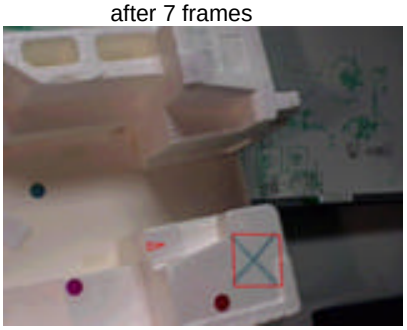
Initial frame



after 7 frames



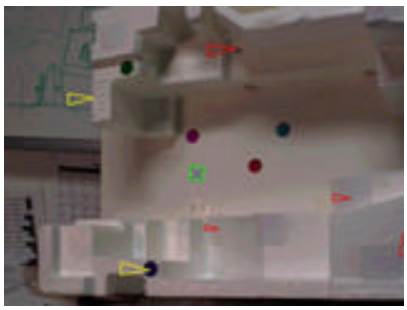
after 58 frames :
Result of EKF with 2 initially unknown features
(red and magenta circles)



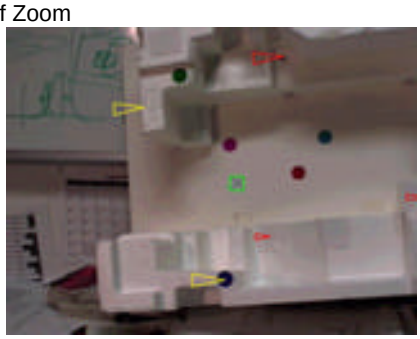
after 58 frames :
Result of RAC with 2 initially unknown features
(red and magenta circles)

Fig. 5 - Sequence depicting a translation of the region of interest

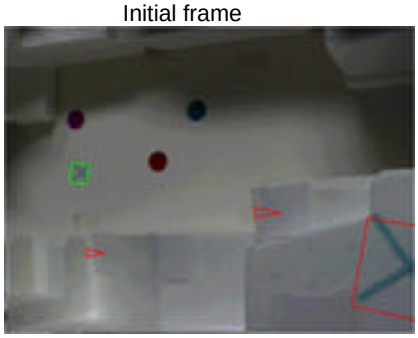
Result of Zoom



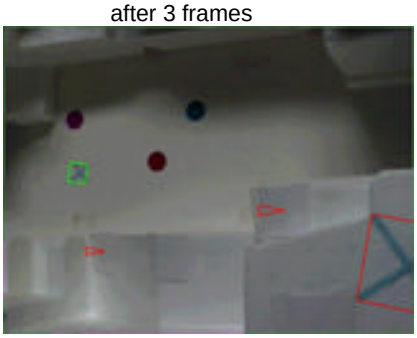
Initial frame



after 3 frames



after 45 frames :
Result of EKF with 2 initially unknown features
(red and magenta circles)

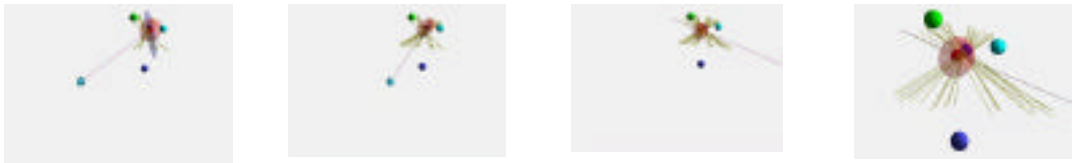


after 45 frames :
Result of RAC with 2 initially unknown features
(red and magenta circles)

Fig. 6 - Sequence depicting a zoom into a smaller regions of interest

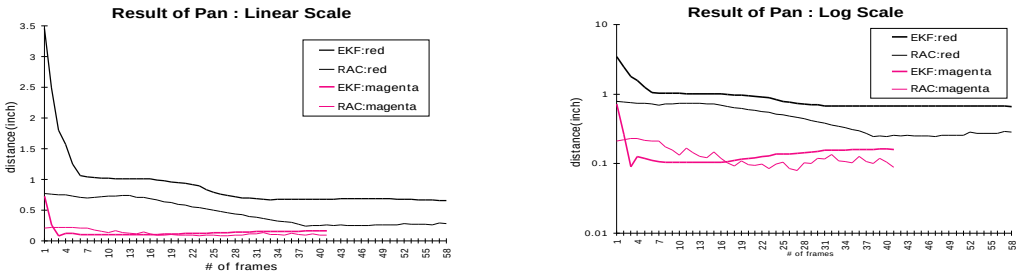
after 13 frames after 28 frames after 38 frames after 43 frames

Fig. 7 - Pan Case: Visualizations of new lines from moving camera views and their effect upon the uncertainty covariance ellipses



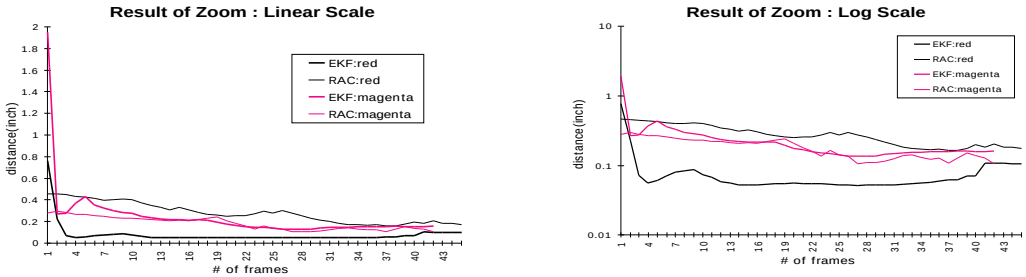
after 19 frames after 29 frames after 40 frames after 40 frames : Close-up

Fig. 8 - Zoom Case: Visualizations of new lines from moving camera views and their effect upon the uncertainty covariance ellipses



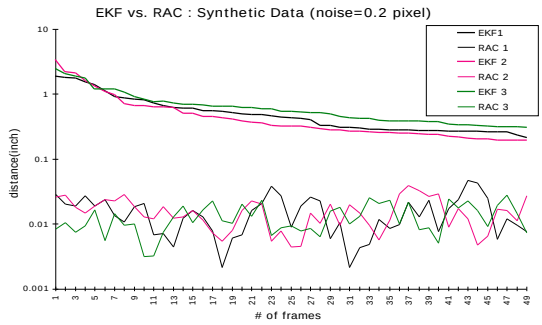
Result of Pan : Linear scale distances Result of Pan : Log scale distances

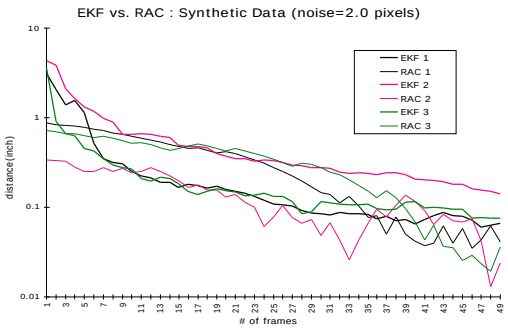
Fig. 9 - Pan Case: Distance between measured and digitized positions of new fiducials



Result of Zoom : Linear scale distance Result of Zoom : Log scale distance

Fig. 10 - Zoom Case: Distance between measured and digitized positions of new fiducials





Synthetic Data(S.D. = 0.2 pixel): Logarithm scale

Synthetic Data(S.D. = 2.0 pixels): Logarithm scale

Fig. 11 - Synthetic Data: Distance between measured and true positions of new fiducials

5.0 References

1 I. Sutherland "A Head-Mounted Three-Dimensional Display," Fall Joint Computer Conference, pp. 757-775, 1968

2 E. Foxlin, "Inertial Head-Tracker Sensor Fusion by a Complementary Separate-Bias Kalman Filter," Proceedings of VRAIS'96, pp. 184-194

3 M. Ghazisadedy, D. Adamczyk, D. J. Sandlin, R. V. Kenyon, T. A. DeFanti, "Ultrasonic Calibration of a Magnetic Tracker in a Virtual Reality Space," Proceedings of VRAIS'95, pp. 179-188

4 D. Kim, S. W. Richards, T. P. Caudell, "An Optical Tracker for Augmented Reality and Wearable Computers," Proceedings of VRAIS'97, pp. 146-150

5 K. Meyer, H. L. Applewhite, F. A. Biocca, "A Survey of Position Trackers," Presence: Teleoperator and Virtual Environments, Vol. 1, No. 2, pp. 173-200, 1992

6 H. Sowizral, J. Barnes, "Tracking Position and Orientation in a Large Volume," Proceedings of IEEE VRAIS'93, pp. 132-139

7 A. State, G. Hirota, D. T. Chen, B. Garrett, M. Livingston, "Superior Augmented Reality Registration by Integrating Landmark Tracking and Magnetic Tracking," Proceedings of Siggraph96, Computer Graphics, pp. 429-438

8 M. Ward, R. Azuma, R. Bennett, S. Gottschalk, H. Fuchs, "A Demonstrated Optical Tracker with Scalable Work Area for Head-Mounted Display Systems," Proceedings of the 1992 Symposium on Interactive 3D Graphics," pp. 43-52

9 G. Welch, G. Bishop, "SCAAT: Incremental Tracking with Incomplete Information," Proceedings of Siggraph97, Computer Graphics, pp. 333-344

10 G. Klinker, K. Ahlers, D. Breem, P. Chevalier, C. Crampton, D. Greer, D. Koller, A. Kramer, E. Rose, M. Tuceryan, R. Whitaker, "Confluence of Computer Vision and Interactive Graphics for Augmented Reality," Presence: Teleoperator and Virtual Environments, Vol. 6, No. 4, pp. 433-451, August 1997

11 S. Feiner, A. Webster, T Krueger III, M MacIntyre, E. Keller, "Architectural Anatomy," Presence: Teleoperator and Virtual Environments, Vol. 4, No. 3, pp. 318-325, Summer 1995

12 T. P. Caudell, D. M. Mizell, "Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes," Proceedings of the Hawaii International Conference on Systems Sciences, pp. 659-669, 1992

13 S. Feiner, B. MacIntyre, D. Seligmann, "Knowledge-Based Augmented Reality," Communications of the ACM, Vol. 36, No. 7, pp 52-62, July 1993

14 E. Natonek, Th. Zimmerman, L. Fluckiger, " Model Based Vision as Feedback for Virtual Reality Robotics Environments," Proceedings of VRAIS'95, pp. 110-117

15 U. Neumann, Y. Cho, "A Self-Tracking Augmented Reality System," Proceedings of ACM Virtual Reality Software and Technology '96, pp. 109-115

16 J. Rekimoto, "NaviCam: A Magnifying Glass Approach to Augmented Reality," Presence: Teleoperator and Virtual Environments, Vol. 6, No. 4, pp. 399-412, August 1997

17 R. Sharma, J. Molineros, "Computer Vision-Based Augmented Reality for Guiding Manual Assembly," Presence: Teleoperator and Virtual Environments, Vol. 6, No. 3, pp. 292-317, June 1997

18 M. Uenohara, T. Kanade, "Vision-Based Object Registration for Real-Time Image Overlay," Proceedings of Computer Vision, Virtual Reality, and Robotics in Medicine, pp. 13-22, 1995

19 T. Starner, S. Mann, B. Rhodes, J. Levine, J. Healey, D. Kirsh, R. Picard, A. Pentland, "Augmented Reality Through Wearable Computing," Presence: Teleoperator and Virtual Environments, Vol. 6, No. 4, pp. 386-398, August 1997

20 K. Kutulakos, J. Vallino, "Affine Object Representations for Calibration-Free Augmented Reality," Proceedings of VRAIS'96, pp. 25-36

21 J. P. Mellor, "Enhanced Reality Visualization in a Surgical Environment," Master's Thesis, Dept. of Electrical Engineering, MIT, 1995

22 T. S. Huang, A. N. Netravali, "Motion and Structure from Feature Correspondences: A Review," Proceedings of the IEEE, Vol. 82, No. 2, pp. 252-268, Feb. 1994

23 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.

24 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

25 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.

26 S. Ganapathy, "Real-Time Motion Tracking Using a Single Camera," AT&T Bell Labs Tech Report 11358-841105-21-TM, Nov. 1984

27 R. Horaud, B. Conio, O. Leboulleux, "An Analytic Solution for the Perspective 4-point Problem," Compter Vision, Graphics, and Image Processing, Vol. 47, No. 1, pp. 33-43, 1989

28 R. Sharma, J. Molineros, "Computer Vision-Based Augmented Reality for Guiding Manual Assembly," Presence: Teleoperator and Virtual Environments, Vol. 6, No. 3, pp. 292-317, June 1997

29 A. Azarbayejani, A. Pentland, "Recursive Estimation of Motion, Structure, and Focal Length," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17, No. 6, June 1995

30 T. J. Broida, S. Chandrashekhar, R. Chellappa, "Recursive Estimation from a Monocular Image Sequence," IEEE Transactions on Aerospace and Electronic Systems, Vol. 26, No. 4, pp. 639-655, July 1990

31 S. Soatto, P. Perona, "Recursive 3D Visual Motion Estimation Using Subspace Constraints," CIT-CDS 94-005, California Institute of Technology, Tech Report, Feb. 1994

32 S-L. Iu, K. W. Rogovin, "Registering Perspective Contours with 3-D Objects without Correspondence, Using Orthogonal Polynomials," Proceedings of VRAIS'96, pp. 37-44

33 Eagle Eye, Kinetic Sciences
http://www.kinetic.bc.ca/eagle_eye.html

34 A. Tremeau, N. Borel, "A Region Growing and Merging Algorithm to Color Segmentation," Pattern Recognition, Vol. 30, No. 7, pp. 1191-1203, 1997

35 M. Uenohara, T. Kanade, "Vision-Based Object Registration for Real-Time Image Overlay," Proceedings of Computer Vision, Virtual Reality, and Robotics in Medicine, pp. 13-22, 1995