EyeTap Video–based Featureless Projective Motion Estimation Assisted by Gyroscopic Tracking

Chris Aimone, Andrej Marjan, and Steve Mann
University of Toronto, 10 King's College Road, Toronto, Canada
Tel. (416) 946-3387; Fax. (416) 971-2326
E-mail: {`aimone, amarjan`}`@eyetap.org, mann@eecg.toronto.edu`

Abstract

This paper proposes a computationally economical method of recovering the projective motion of head mounted cameras or EyeTap devices, for use in wearable computer mediated reality. The tracking system combines featureless vision and inertial tracking in a closed loop system to achieve accurate robust head tracking using inexpensive uncalibrated sensors. The combination of inertial and vision techniques provides the high accuracy visual registration needed for fitting computer graphics onto real images and robustness to large interframe camera motion due to fast head rotations. Operating on a 1.2 Ghz Pentium III wearable computer, the system is able to register live video images with less than 2 pixels of error (0.3 degrees) at 12 frames per second.

Keywords: Video Head Tracking, Inertial Head Tracking, Camera Egomotion, Drift Correction, Personal Imaging, Augmented Reality, Mediated Reality, Eyetap

1 Introduction

The EyeTap (see Figure 1) is a wearable computer device that enables the user's vision to be computationally processed and modified in real-time [1].

An EyeTap incorporates a camera and a display arranged with appropriate optical geometry such that the camera and eye have equivalent optic centers and the display accurately resynthesizes the incoming light. This arrangement eliminates parallax between the camera and eye, thus allowing for the seamless addition of virtual information, even when the EyeTap device has been manufactured with partial optical transparency. Optical transparency is often desired for partial mediation, allowing the user to see the majority of their environment with natural light, rather then completely virtual.

Figure 1: Two examples of EyeTap devices that tap the right eye. Note that both eyes can be tapped if stereo computer mediated reality is desired.

The key problem in augmented and mediated reality is that of achieving geometric registration between the real and computer-generated worlds. This is necessary for ensuring that virtual objects are inserted correctly so that they appear to exist in the real world.

One major approach to the registration problem is vision-based tracking, which estimates the user's position and orientation from images captured by a wearable camera. This approach can result in extremely good registration with negligible drift. Pure vision tracking that is unconstrained to controlled environments is very computationally expensive, inhibiting its use as a real–time registration technique on current wearable computers.

In order to decrease the computational complexity of unconstrained vision tracking to managable levels and to increase tracking robustness, a number of researchers have used hybrid tracking schemes that combine inertial tracking devices with vision techniques [2] [3].

Satoh et al [2] proposed a hybrid scheme for three–degrees–of–freedom (3DOF) rotation tracking using precision fiber optic gyroscopes with vision based drift compensation. The system relies on the gyroscope for video rate tracking, and uses a slower natural feature tracking algorithm to compensate for drift.

This paper proposes a similar system that uses inexpensive vibrating element gyroscopes for high frame rate tracking, and a vision tracking algorithm for low frame rate drift compensation. In contrast to the method proposed by Satoh, the system uses featureless vision tracking and has a closed loop structure that allows the mechanically based inertial sensors to be continuously calibrated during use. It tracks the video motion by estimating the projective transformation between video frames. This characteristic is useful for content replacement schemes because it is less constrained than conventional 3DOF rotation tracking. However, for the purpose of estimating physical head position the system is limited to relative 3DOF rotation since does not incorporate and absolute measuring techniques such as an inclinometer or a compass.

2 Head Tracking

The proposed head tracking system combines a featureless vision based tracking algorithm called VideoOrbits[1] [1] with two small, low cost, vibrating element gyroscopic sensors. The system has a closed loop form that enables the calibration of the inertial sensor during use and adds robustness to independently moving objects in the field of view.

VideoOrbits finds best fit Projective Coordinate Transformations (PCTs) to register pairs of overlapping images or scene content (Equation (1)).

$$\mathbf{x}' = \begin{bmatrix} x' \\ y' \end{bmatrix} = \frac{\mathbf{A}[x,y]^T + \mathbf{b}}{\mathbf{c}^T[x,y]^T + 1} = \frac{\mathbf{A}\mathbf{x} + \mathbf{b}}{\mathbf{c}^T\mathbf{x} + 1} \qquad (1)$$

where $\mathbf{A}, \mathbf{b}, \mathbf{c}$ are the eight parameters of the projective coordinate transformation (PCT).

This set of transformations, when applied across the entire image, can describe exactly the relation between two images where all the objects in the scene are static and the camera is only free to rotate and zoom. It is also exact for a planar scene imaged from arbitrary locations. The motion of a head mounted camera or EyeTap between successive frames of video can be approximated very well by a pure rotation, since we often scan our environment using head rotations. This allows VideoOrbits to provide extremely good registration between EyeTap video frames [1], and forms the basis for tracking physical head rotations from a real-time video sequence. Its effectiveness in registration has also been demonstrated in mediated reality content replacement applications [1] for arbitrary camera movement. Unfortunately this method is very computationally intensive. Moreover, complexity increases with interframe displacement, thus inhibiting its real–time implementation on a typical wearable computer, unless the camera motion is kept very small. Figure 2 shows the tracking error in VideoOrbits when the execution time of the algorithm is limited. We can see that if the interframe displacement is kept to less than about four degrees, VideoOrbits converges and the tracking error is in the sub–pixel range (with the camera used, 1 pixel $\approx$ 0.15 degrees). The convergence range is dependent on the scene as well as the computation time. The four degree range was found to be typical for an office environment with a 0.3 s VideoOrbits execution time. The four degree range can be dramatically improved by increasing the execution time as well as combining the method with a block matching scheme such as FFT phase correlation.

To make real–time tracking using VideoOrbits computationally feasible, the proposed system uses a small gyroscope to aid VideoOrbits in finding interframe projective coordinate transformations (PCTs), accelerating the algorithm and improving its ability to handle large interframe displacements due to rapid head movements. Figure 2 shows the gyroscope tracking error. With the gyroscope used alone, we can see that in the zero to four degree range the error is much larger in comparison to VideoOrbits. Beyond four degrees, VideoOrbits does not converge giving extremely large errors while the error in the gyroscope stays fairly constant. By using the gyroscope to estimate the PCT between frames, VideoOrbits is able to converge to a PCT solution giving sub–pixel tracking errors even when there are very large displacements between images.

Although the gyroscope estimate reduces the necessary execution time of VideoOrbits, the algorithm cannot be run at more that a few frames per second on current wearable computers. To achieve higher frame rates, the gyroscope is used to estimate the camera motion between VideoOrbits solutions. This reduces the overall accuracy, but allows the frame rate to be increased to usable level.

The inertial tracking device used is a pair of small, low cost vibrating element gyroscopes made by Gyration Inc. The manufacturer claims a maximum accuracy of 0.15 deg/s with proper calibration and drift correction. Due to the electromechanical nature of the system, the calibration parameters are affected by temperature and power supply fluctuations and the gradual change in gyroscope characteristics over time. The proposed system aims to solve this problem through continuous closed loop calibration.

---

[1] VideoOrbits available for free at www.wearcam.org/orbits/

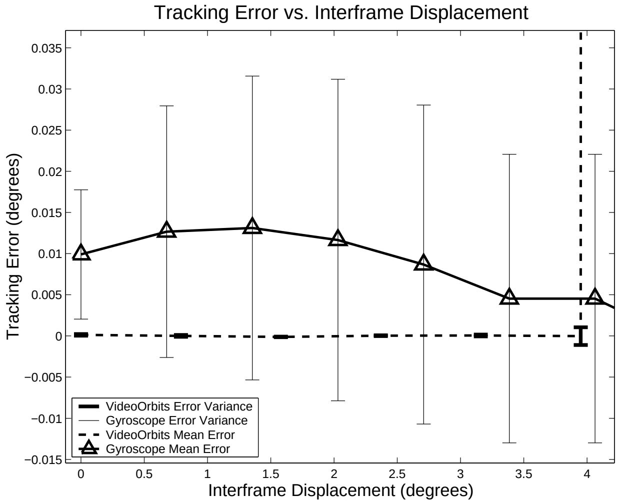

Figure 2: This graph compares the tracking ability of VideoOrbits and the gyroscope used. The error bars represent the variance in the tracking error and are displayed as one tenth the actual size. For this experiment, the execution time for VideoOrbits was limited to 0.3 s per frame. Under these circumstances we can see that if the interframe displacement is less than four degrees, the algorithm converges, resulting in very small error. Beyond four degrees, convergence in not achieved, resulting in unusable PCT estimates. While not shown, the error in the gyroscope remains fairly constant beyond four degrees of interframe displacement.

2.1 Gyroscopic/VideoOrbits Tracking System

The Gyroscope/VideoOrbits System (GVS) tracks the absolute position of a camera in terms of the projective coordinate transformation (PCT) between the current frame of video and a base frame. In terms of camera rotation, the base frame would exist at $(\theta, \psi, \phi) = (0, 0, 0)$, where θ, ψ and ϕ are the Euler angles describing the direction of the camera's optical axis. Tracking the camera motion as a PCT makes sense since the primary goal of head tracking in augmented reality systems is visual registration. If the camera undergoes pure physical 3-D rotation about its optic center, the rotation can be recovered from the PCT provided the camera intrinsic parameters are known. Since in many scenarios, the user's head motion can be approximated very well by pure rotation, it makes it possible to estimate the rotational position of the camera with respect to its visual environment from its PCT position.

The GVS is a closed loop system in which a gyroscope provides an estimate of camera motion to allow VideoOrbits to determine the PCT between video frames even in the presence of large interframe camera movements. The high accuracy PCTs produced by VideoOrbits are used to build a linear model for the gyroscope in order to correct for gyro drift and scaling problems. A block diagram of the system is shown in Figure 3.

Absolute Tracking with respect to the visual environment is measured by composing the relative PCTs between video frames. To eliminate the effects of cumulative error a spherical map of the environment is continuously generated as the user wears the device. The continuous regeneration of this map allows the system to maintain good visual registration even when head motion is not limited to pure rotation. Under these circumstances, the achieved accuracy depends on the amount of translational camera motion and the distance to the objects in the user's field of view.

Video Capture and Undistortion

Digital video is provided by an inexpensive IEEE1394 webcam. The IEEE1394 format allows programmatic control of exposure settings for most cameras, which is very desirable since VideoOrbits uses the brightness constancy constraint. Image brightness information is obtained by selecting the luminance component of the YUV color space. It is known that the YUV color space yields near optimal energy compaction (most uncorrelated) in most color images with a single luminance component and two chrominance components. Camera barrel distortion was removed from the camera images with an optimized undistortion function in the Intel Open Source Computer Vision Library (OpenCV).

Acquire and process Gyroscope signals

A PIC microcontroller was used to digitize the analog rate outputs from the gyroscope and to compute the interframe integration. Rotation information is provided on demand to the wearable computer over a RS232 communication port. While gyro drift is inevitable, it remains relatively constant and can be subtracted from the rate output fairly well. The gyroscope output is also scaled to match the units used in the system:

$$\begin{bmatrix} \theta_g \\ \psi_g \\ \phi_g \end{bmatrix} = \begin{bmatrix} \mathbf{G_1} \end{bmatrix} \left(\begin{bmatrix} \theta_{raw} \\ \psi_{raw} \\ \phi_{raw} \end{bmatrix} - \begin{bmatrix} \mathbf{G_0} \end{bmatrix} \right) \qquad (2)$$

where $[\theta_g \psi_g \phi_g]^T$ are the gyroscope rotation angles in degrees and $[\theta_{raw} \psi_{raw} \phi_{raw}]^T$ are the raw signals received from the PIC. G_0 and G_1 are the vectors for correcting drift and scaling respectively.

Unfortunately as with all analog systems, component values drift with temperature, and load changes can affect the power system levels. Both of these effects will cause short term variations in the required drift and scaling parameters. This problem was solved by continuous re–estimation of parameters G_0 and G_1 to adapt to local statistics.

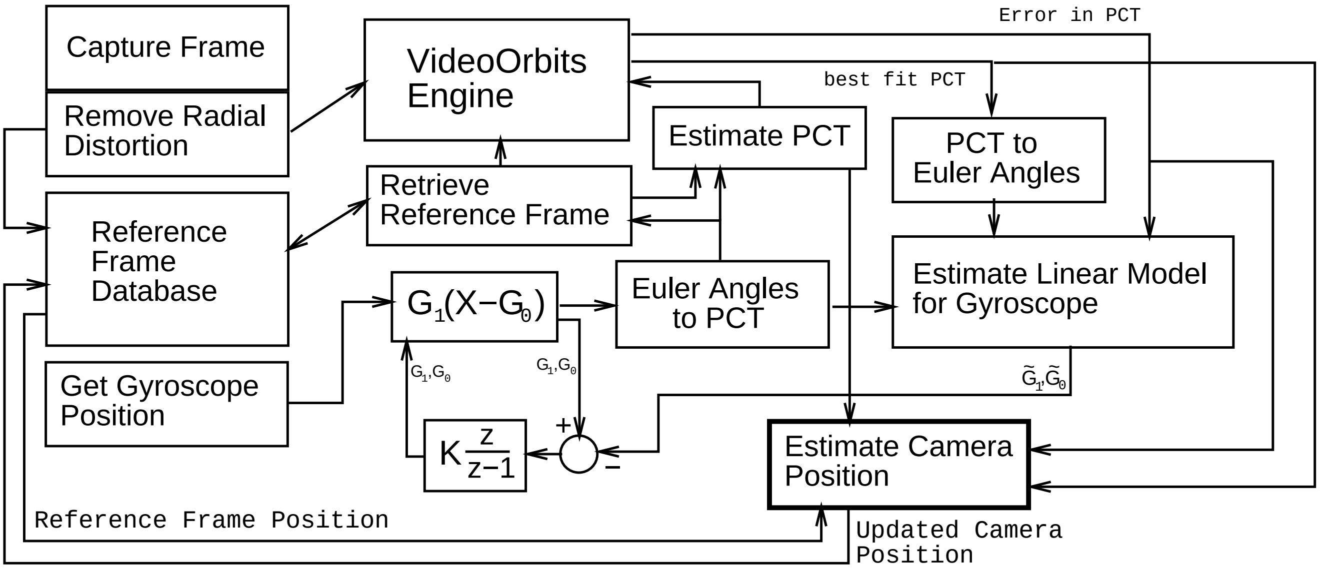

Figure 3: Gyro / VideoOrbits Tracker Block Diagram

Calculate Equivalent Projective Transformation from Euler angles

In order for the gyroscope for provide an initial transformation estimate for VideoOrbits, the interframe rotation experienced by the gyroscope must be converted into a PCT. To calculate this PCT, a rotation matrix is formed using the relative Euler Angles returned by the gyroscope. This matrix is composed with a fixed 3-D rotation matrix to align the gyroscope's coordinate system with the camera coordinate system. VideoOrbits operates on images that have been scaled such that the upper left corner of the image is $(0, 0)$ and the bottom right corner is $(1, 1)$. To create a PCT from the obtained relative rotation matrix an appropriate change of coordinates is applied:

$$\mathbf{W}\mathbf{R}\mathbf{W}^{-1} = \lambda \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & 1 \end{bmatrix} = \lambda P_g \tag{3}$$

$$\text{where } \mathbf{W} = \begin{bmatrix} \frac{f_x}{S_x} & 0 & \frac{O_x}{S_x} \\ 0 & \frac{f_y}{S_y} & \frac{O_y}{S_y} \\ 0 & 0 & 1 \end{bmatrix}, \tag{4}$$

$\mathbf{R}$ is the rotation matrix corresponding to the gyroscope rotations in the camera coordinate system, f_x and f_y are the camera focal lengths in the horizontal and vertical directions, S_x and S_y are the dimensions of the video images in the horizontal and vertical directions, (O_x, O_y) is the point of intersection of the optical axis of the camera and the sensor array (in units of pixels), and λ is a scale factor used to force the last entry in the PCT matrix to be one. $\mathbf{A}, \mathbf{b}$ and $\mathbf{c}$ are the PCT parameters defined in in Equation (1).

Reference Frame Database

This system tracks the absolute camera motion by finding the PCT (P_{abs}) relating the current frame of video to a base frame. One method of determining P_{abs} is to compose the relative PCTs, P_{n-1} and P_n, from temporally adjacent frames, numbered $n-1$ and n:

$$P_{abs} = \prod_{n=1}^{N} P_n \tag{5}$$

where P_{abs} and P_n are expressed in matrix form:

$$P = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^T & 1 \end{bmatrix} \tag{6}$$

This method however is quite prone to drift since the errors in the relative PCTs have a cumulative effect on the absolute position. In this system the cumulative error is reduced by storing well registered images as reference frames. Absolute rotation is then calculated by composing the relative PCTs (P_n) relating the current video frame to a reference frame, with the PCT (P_r) relating the reference frame to the base frame. The resulting performance increase can be seen in Figure 4.

The Reference Frame Database is a spherically indexed database of reference images of the environment that are continuously collected and updated through normal system use. Each image in the database includes a time stamp, a PCT (P_r) describing its position

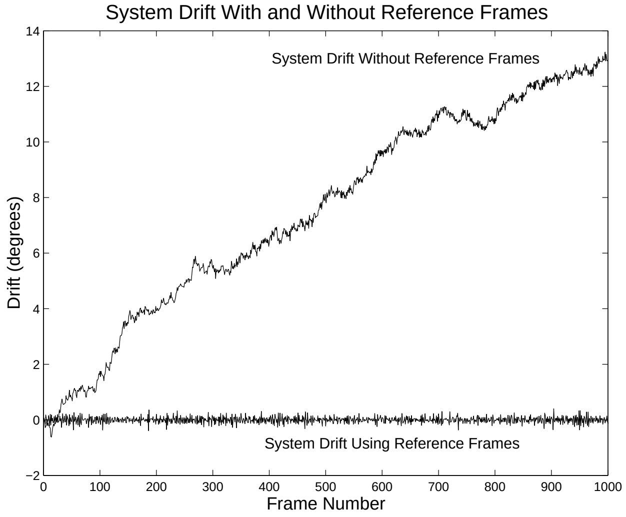

Figure 4: This graph shows the effect of reference frame use on system drift. For this experiment, the camera was held in a fixed position.

with respect to the base frame, and a PCT certainty measure to allow for judicious reference frame selection. Reference frames are selected for use with VideoOrbits based on the current position estimate generated by the gyroscope.

It is important to note that if large head rotations occur, it is possible for consecutive image frames to have no overlap at all. This can prevent VideoOrbits from estimating a useful PCT in the case where the sphere of environment images is poorly populated and no nearby reference images exist In this case, a new reference image will be stored with a gyro-generated PCT. New reference images with gyroscope generated PCTs can also be created if VideoOrbits cannot find good registration with an existing reference frame.

Images whose associated PCTs were generated by VideoOrbits are selected in preference to those whose positions were estimated by the gyroscope alone. More recent images are selected in preference to older images to account for environment changes. Each time a reference frame is used successfully, its timestamp is updated to ensure that it continues to be used instead of its neighbors.

Reference frames are saved when the current video image is offset from the current reference frame by a significant amount. This amount is dependent on the camera's field of view and the system resources of the operating platform. The sphere of reference images is mapped to a two dimensional matrix, indexed by equal increments of azimuth and elevation. In experimentation, four degree increments were used. Pinching at the poles of the sphere is eliminated by dividing the matrix into petals of valid image locations. The remaining entries of the matrix are marked as NULL, as they are redundant. Searching this matrix is simple since the camera position can be directly related to the position in this matrix. If the search includes NULL entries, their nearest neighbors (of greater azimuth and elevation) are used. Near the poles, searches tend to wrap to opposing sides of the location matrix. It is important to note that frames are saved only when the estimated camera velocity at the time of image capture is small enough to produce acceptable amounts of image blurring.

Estimate PCT

The gyroscope estimated PCT is used to aid VideoOrbits in registering the current video frame to a reference frame. Since the PCT produced directly by the gyroscope measurement is relative to the last frame of video, it must be transformed to be relative to the reference frame that VideoOrbits will use: If P_g is the gyroscope estimated PCT, P_l is the PCT describing the position of the last frame of video and P_r is the describing the position of the reference frame, the estimated PCT between current frame of video and the reference frame (P_e) is given by:

$$P_e = \lambda P_r^{-1} P_l P_g \tag{7}$$

where P_e, P_r, P_l, P_g are PCTs in the matrix form. As before, λ is used to force the $(3,3)$ entry of P_e to be 1.

VideoOrbits Engine

VideoOrbits estimates the PCT (P_o) between the current frame of video and the reference frame by substituting $u_m = \frac{Ax+b}{c^T x+1} - x$ into the Horn and Schunk brightness constancy constraint equation (BCCE) [4]. This method is known as 'projective–flow' [5]. Because of effects such as parallax in the images due to motion other than rotation, independently moving objects in the scene, and camera blur, the PCT cannot fit perfectly at all points of the images. Thus the solution we seek is the best estimate of the PCT parameters in a least squares sense across all image points. This is solved by minimizing:

$$\begin{aligned} \varepsilon_{flow} &= \sum \left(\mathbf{u_m}^T \mathbf{E_x} + E_t\right)^2 && (8)\\ &= \sum \left(\left(\frac{\mathbf{Ax}+\mathbf{b}}{\mathbf{c}^T\mathbf{x}+1} - \mathbf{x}\right)^T \mathbf{E_x} + E_t \right)^2 && (9) \end{aligned}$$

where E_x is the 2-D spatial gradient of the image $[d_x, d_y]$ at some point (x, y), and E_t is the temporal gradient, representing the change in brightness at a particular pixel coordinate from one image to the next.

Minimizing (9) is simplified by weighting by $(\mathbf{c}^{\mathbf{T}}\mathbf{x} + 1)$, giving:

$$\varepsilon_w = \sum \left((\mathbf{A}\mathbf{x} + \mathbf{b} - (\mathbf{c}^T\mathbf{x} + 1)\mathbf{x})^T \mathbf{E}_{\mathbf{x}} + (\mathbf{c}^T\mathbf{x} + 1)E_t\right)^2 \tag{10}$$

where ε_w denotes the weighted error.

To solve for the minimum we differentiate with respect to the free parameters $\mathbf{A}, \mathbf{b},$ and $\mathbf{c}$, and set the result to zero to give a linear solution:

$$\left(\sum \phi\phi^T\right)[a_{11}, a_{12}, b_1, a_{21}, a_{22}, b_2, c_1, c_2]^T \tag{11}$$

where $\phi^T = [E_x(x,y,1), E_y(x,y,1), xE_t - x^2E_x - xyE_y, yE_t - xyE_x - y^2E_y]$.

Since only the first derivative of the spatial gradient is used in the BCCE, the modeled change in pixel brightness due to translational velocity is only accurate when the motion between successive image is small. The set of PCTs forms a group, allowing VideoOrbits to utilize the group law of composition in a multiscale iterative technique to find the optimal PCT between two images. The iterations proceed as follows:

(1) Estimate the projective coordinate transformation $\mathbf{P}$

(2) Use this $\mathbf{P}$ to transform the appropriate original image

(3) Estimate another PCT, $\mathbf{P_2}$, between the transformed image and the other original image.

(4) Generate an improved $\mathbf{P}$ by composing $\mathbf{P_2}$ with $\mathbf{P}$

(5) goto (2) until the desired accuracy has been achieved

This scheme allows the final iterations to estimate the PCT between two images that have very little motion between them, thus allowing the first order approximation of pixel brightness change to be sufficiently accurate.

VideoOrbits uses the PCT estimate P_g from the gyroscope as a starting point (step (1)) for the iterative scheme described. VideoOrbits runs for a prescribed number of iterations depending on the computational power of the wearable computer, and returns its best estimate of the PCT parameters. VideoOrbits also returns a Mean Squared Error (MSE) value for the returned PCT, which is used as a measure of registration accuracy. The MSE is calculated by adding the squared difference between the original image and its transformed temporal neighbor. Unfortunately the MSE measure is very sensitive to the nature of the images used, and thus sensitive to the user's local environment. For example, well registered images in a dim environment will have lower MSE values than in a bright environment. To cope with this, local statistics are kept on the MSE levels to provide intelligent adaptation to changing environments.

Estimate Euler Angles from the PCT

In order to close the loop and allow VideoOrbits to correct the gyroscope scaling and drift parameters, an equivalent 3-axis rotation must be obtained from the PCT. It is known that the PCT has eight free scalar parameters and is thus not limited to pure rotation, so it is necessary to find the best fit 3-D rotation matrix and have some indication of the error to determine if a rotation is valid and can be used in the statistical model for the gyroscope. To find the required rotations, a PCT (P_{rel}) describing the relative rotation since the last successful execution of VideoOrbits is found: $P_{rel} = P_r^{-1}P_{n-1}P_o$, where P_r, P_{n-1}, and P_n are the PCTs relating the reference, previous, and current video frames to the base frame. This relative PCT is then approximated by a pure rotation matrix using a singular value decomposition (SVD) of P_{rel}. The SVD of P_{rel} is given by:

$$P_{rel} = USV^T \tag{12}$$

where S is a diagonal matrix of singular values. The best fit rotation matrix R in terms of the Frobenius norm between itself and P_{rel} is given by:

$$R = UV^T \tag{13}$$

This rotation matrix is then used to compute the corresponding Euler angles $[\theta_o\psi_o\phi_o]^T$. This computation is not difficult, but is lengthy and thus will not be presented here.

The Frobenius norm is calculated as follows, to give an error measure in the rotation matrix:

$$||R - P_{rel}||_F^2 = Tr((R - P_{rel})^T(R - P_{rel})) \tag{14}$$

This measure is used to help identify non-convergent PCT estimates from VideoOrbits and select Euler angle estimates from P_{rel} that are suitable for use in gyroscope drift and scaling correction.

Estimate G_1 and G_0

In order to estimate the gyroscope drift and scaling parameters, a buffer of recent data points correlating the gyroscope Euler angles $[\theta_g\psi_g\phi_g]^T$ to the VideoOrbits Euler angles $[\theta_o\psi_o\phi_o]^T$ is maintained. Data points

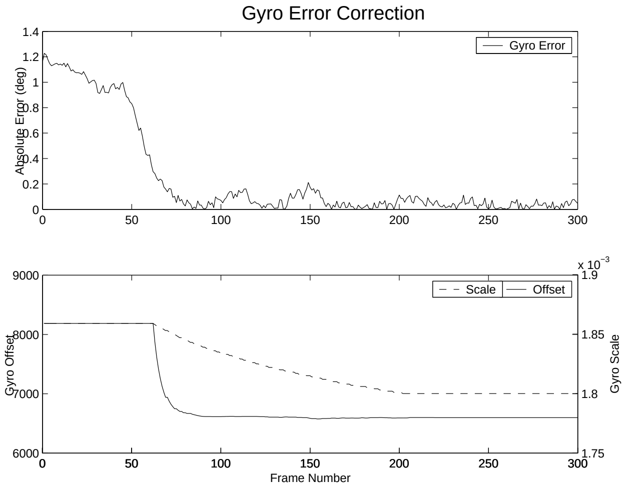

Figure 5: The above data were taken in a scenario where power supply voltage to the gyro had been changed (At frame 0) to simulate power level fluctuations in a battery–driven system. At around frame 65, the error in the gyro model dropped below a threshold value and the system started to correct the calibration parameters. We can see by the reduction of the absolute error that the system has converged to a more desirable state.

associated with poor MSE with respect to the local statistics or with high error in the rotation matrix approximation of P_{rel} are rejected. These data points are used to create a linear model, using linear regression, that describes both the drift and scaling parameters of the gyroscope. For this model, the standard error in the slope and the intercept, as well as the correlation coefficient are calculated. These statistics are used to decide on corrective action for the parameters. The stringency of the decision tolerances increases as gyroscope calibration improves to ensure convergence to the optimal solution. To allow re–calibration in the event of a large shift in parameter values during operation, the decision tolerances are relaxed if a good linear fit is found with significantly different parameters. Large shifts in parameter values were found to occur when the system was exposed to large temperature changes such as when moving from an indoor to an outdoor environment.

3 Results: System Evaluation

The system presented has shown the ability to correct for the gyro scaling and drift parameters through normal wear. The length of time required for calibration depends on the user's motions, since VideoOrbits needs to achieve good registration between frames to contribute meaningful data to the gyro model. Figure 5 demonstrates the success of the closed-loop correction of gyro drift and scaling.

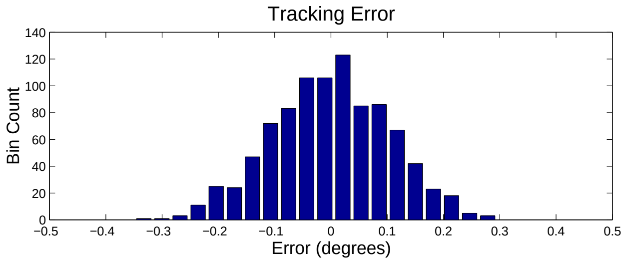

Figure 6: The histogram shows the distribution error in real–time image registration with a well populated reference frame database.

3.1 Tracking Performance

Operating on a 1.2 Ghz Pentium III (With 512 MByte RAM and running Linux 2.4.18), the system was able to register images with less than 2 pixels of error (0.3 degrees) at 12 frames per second. Figure 6 shows the distribution of registration errors with the system running in a static environment with a well populated reference frame database. These results apply for conditions where the camera motion can be approximated well by 3DOF rotations. The absolute tracking performance of the system however, suffers greatly when it is exposed to large accelerations such as running or jumping. It is also sensitive to low light conditions, lower shutter speeds cause blurry images through low shutter speeds.

Figure 7 shows the tracking of a point in the scene with significant head motion. This figure also shows how the tracking algorithm is robust in the presence of independently moving objects. VideoOrbits alone is subject to very large errors when an independently moving object is introduced. However, with the gyroscope in the system, the PCT parameters produced by VideoOrbits are rejected due to high associated MSE values.

Figure 8 illustrates real–time registration results. In this example the frame rate has been reduced to 3 frames per second, to produce large interframe displacements and show the system's ability to cope with large head motions.

4 Conclusion

The hybrid Gyro / VideoOrbits System (GVS) operates in real–time, with reasonable frame rates on currently available wearable computers. It has demonstrated video image registration with sufficient accuracy to be used in many augmented reality applications where the motion of an eyetap device or head–mounted camera can be approximated by 3DOF rotation. It is not dependent on specialized environments, but it performs best in well lit static environments. The GVS has already been successfully used in an augmented re-

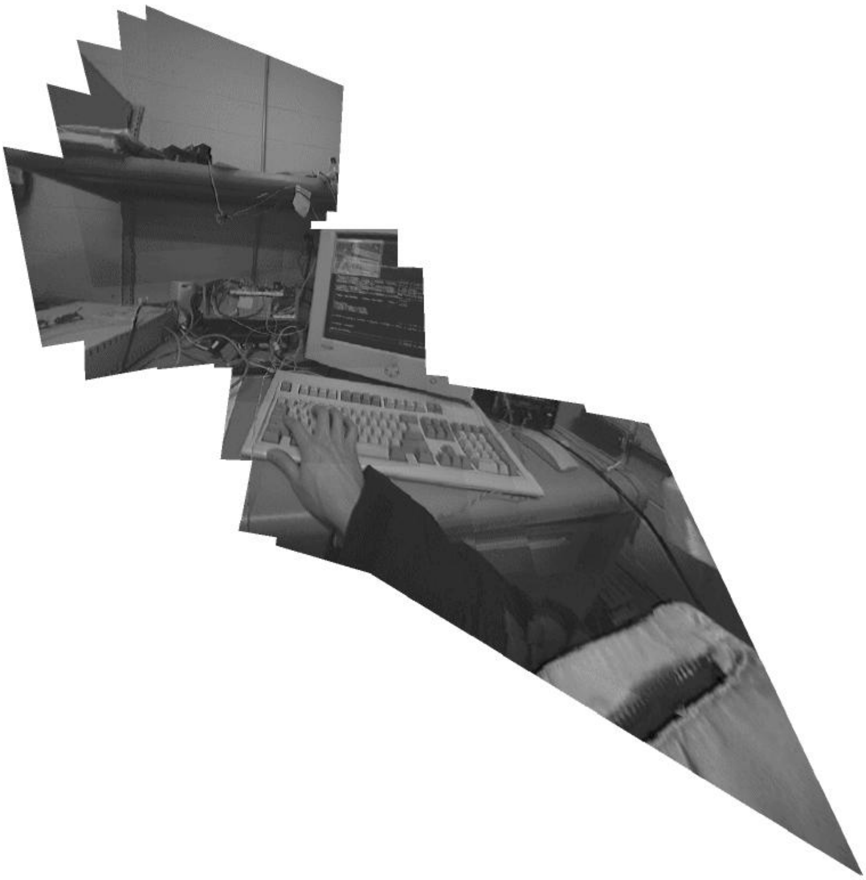

Figure 8: Real–time Image registration in a close environment. In this instance the frame rate has been reduced to 3 frames per second, to produce extra large interframe displacements and put the gyroscope to the test.

Figure 7: The following images show the GVS tracking a point with significant head rotation in a near environment. The images containing the hand, demonstrate the system's ability to track even when blocked by a large independently moving object.

ality application that locates and tracks visible wireless devices in the user's environment. The GVS tracking solution uses small, low cost, 'off the shelf' parts and thus promises to help bring augmented reality to the general computer user.

References

[1] Steve Mann. *Intelligent Image Processing.* John Wiley and Sons, November 2 2001. ISBN: 0-471-40637-6.

[2] K. Satoh, M. Anabuki, H. Yamamoto, and H. Tamura. A hybrid registration method for outdoor augmented reality. In *Proceedings of International Symposium on Augmented Reality (ISAR2001)*, 2001.

[3] S. You, U. Neumann, and R. Azuma. Hybrid inertial and vision tracking for for augmented reality registration. In *Proceedings of IEEE VR 2000, pp. 260-267, 2000.*, 2000.

[4] B. Horn and B. Schunk. Determining Optical Flow. *Artificial Intelligence*, 17:185–203, 1981.

[5] Steve Mann. Humanistic intelligence/humanistic computing: 'wearcomp' as a new framework for intelligent signal processing. *Proceedings of the IEEE*, 86(11):2123–2151+cover, Nov 1998. http://wearcam.org/procieee.htm.