Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

An Empirical Study into the Robustness of Split Covariance Addition (SCA) for Human Motion Tracking

Simon J. Julier and Joseph J. LaViola Jr.

Abstract— **Accurate human body tracking is extremely important for many virtual and augmented reality systems. However, tracking human motion is extremely difficult. Some of the difficulties arise from the fact that accurate process models of human motion are difficult to derive. Approximate models can have substantial time-correlated process noise terms. In this paper we examine the effectiveness of using the Split Covariance Addition (SCA) algorithm as part of a human head orientation estimation system. We perform a series of empirical experiments to compare the performance of several implementations of SCA with an Extended Kalman Filter (EKF). The results suggest that the benefits of SCA are mixed. It leads to filters which are slightly more robust and have slightly more accurate angular velocity estimates than the EKF. However, the absolute orientation estimate is slightly worse than the EKF.**

I. Introduction

Human body tracking is extremely important for many virtual and augmented reality systems [1]. Accurate head tracking is used to calculate the viewpoint from which the graphics are drawn. Other parts of a user's body (such as wrists and hands) are often tracked to permit the use of 3D interaction devices [2]. However, people are extremely sensitive to errors in tracking. For example, latency in estimating head orientation of only a few milliseconds can at best be distracting and at worst lead to cybersickness [3]. Similarly, errors in tracking of the other parts of a human body can make the system difficult to interact with and use effectively [4].

The solution to these problems is to use an appropriate estimation algorithm which can be used to filter noise from tracking systems and predict the future state to reduce the effects of latency [5]. However, developing a suitable estimator can be extremely difficult. Head motion, for example, often consists of long periods of relative inactivity punctuated by short bursts of violent activity (angular accelerations can exceed $600^{\circ}\mathrm{s}^{-2}$) [5]. Motions often contain substantial time correlations which can degrade the filter's performance.

These problems have been partially overcome through the use of Multiple Model Adaptive Estimation (MMAE) [6, 7]. A bank of filters are implemented where each filter uses a process model which corresponds to a particular class of head motion (such as head is still, head is turning at constant angular velocity, head is turning at constant angular acceleration). Each filter operates independently and in parallel with all other filters. Probabilistic decision rules, based on the likelihood of the innovation, are used to evaluate the probability that, at any given time, a particular model is correct. Although Kyger [6] reports that MMAE schemes yield better estimates than single prediction schemes, each filter is still implemented under the assumption that the process noise terms act independently. However, because no filter is truly an accurate description of the trajectory, time correlated errors are still introduced and degrade the performance of the filter.

To address the issue of prediction in systems with correlated noise terms, an algorithm known as Split Covariance Addition (SCA) has been developed [8]. Given two random variables whose means and covariances are known, but have unknown correlations, SCA calculates the smallest covariance which is guaranteed to be consistent. Therefore, SCA has the potential to address many of the issues concerned with human motion tracking. However, no empirical analysis of the performance of SCA has been conducted.

This paper performs the first empirical study on the use of SCA to help refine the estimates of human motion. Although there are many different types of motion that could be examined, we focus on head orientation because of its importance in providing visual fidelity [9]. The study suggests that, compared with an Extended Kalman Filter (EKF), the benefits of using SCA are mixed and marginal. Although the resulting estimator is slightly more robust and has slightly more accurate angular velocity estimates, the absolute orientation estimate is slightly worse than that of an EKF. However, given the limitations of the study we believe the results indicate that more research is warranted.

II. The EKF and SCA Algorithms

A. System Description

The system is described by the equations

$$\mathbf{x}(k) = \mathbf{f}[\mathbf{x}(k-1), \mathbf{u}(k-1), \mathbf{v}(k-1), k-1]$$
$$\mathbf{z}(k) = \mathbf{h}[\mathbf{x}(k), \mathbf{u}(k-1), \mathbf{w}(k), k-1],$$

where $\mathbf{x}(k)$ is the state vector, $\mathbf{u}(k-1)$ is the control input, $\mathbf{v}(k-1)$ is the process noise, and $\mathbf{f}[\cdot,\cdot,\cdot,\cdot]$ is the discrete time state transition equation, $\mathbf{z}(k)$ is the observation vector, $\mathbf{w}(k)$ the observation noise vector and $\mathbf{h}[\cdot,\cdot,\cdot,\cdot]$ the discrete time observation equation. The noise vectors $\mathbf{v}(k-1)$ and $\mathbf{w}(k)$ are *assumed* to zero-mean and uncorrelated with covariances $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ respectively. The validity of this assumption will be considered later.

S. Julier is with Naval Research Laboratory / ITT AES, 4555 Overlook Avenue SW, Washington, DC 20375 Email: julier@ait.nrl.navy.mil

J. LaViola is with Brown University, Department of Computer Science, Box 1910 Providence, RI 02912 USA Email: jjl@cs.brown.edu

0-7803-8335-4/04/$17.00 ©2004 AACC

The purpose of the estimator is to generate a *consistent* estimate of $\mathbf{x}(i)$. Using the notation from [10], the estimate of $\mathbf{x}(i)$ using all observations up to time j is $\hat{\mathbf{x}}(i \mid j)$ with covariance $\mathbf{P}(i \mid j)$. Let $\tilde{\mathbf{x}}(i \mid j) = \mathbf{x}(i) - \hat{\mathbf{x}}(i \mid j)$. The estimate is consistent if

$$\mathbf{P}(i \mid j) - \mathrm{E}\left[\tilde{\mathbf{x}}(i \mid j)\, \tilde{\mathbf{x}}^T(i \mid j)\right] \geq \mathbf{0}$$

where $\geq \mathbf{0}$ means that the difference between the two matrices is positive semidefinite. In other words, the filter should never under estimate the mean squared error in the estimate.

B. Extended Kalman Filter

The EKF utilizes two steps: prediction followed by update.

In the prediction step, the estimate at time step $k-1$ is projected to the current time k,

$$\hat{\mathbf{x}}(k \mid k-1) = \mathbf{f}\left[\hat{\mathbf{x}}(k-1 \mid k-1), \mathbf{u}(k), \mathbf{0}\right] \quad (1)$$

$$\mathbf{P}(k \mid k-1) = \mathbf{\Phi}_F(k)\, \mathbf{P}(k-1 \mid k-1)\, \mathbf{\Phi}_F(k) + \mathbf{Q}(k-1), \quad (2)$$

where $\mathbf{\Phi}_F(k)$ is the (linearized) transformation matrix.

The update step combines the measurement with the prediction using a minimum mean squared error linear update rule,

$$\hat{\mathbf{x}}(k \mid k) = \hat{\mathbf{x}}(k \mid k-1) + \mathbf{W}(k)\, \boldsymbol{\nu}(k) \quad (3)$$

$$\mathbf{P}(k \mid k) = \mathbf{X}(k)\, \mathbf{P}(k \mid k-1)\, \mathbf{X}^T(k) + \mathbf{W}(k)\, \mathbf{R}(k)\, \mathbf{W}^T(k)$$

where

$$\mathbf{C}(k) = \mathbf{P}(k \mid k-1)\, \boldsymbol{\nabla}^T\mathbf{h} \quad (4)$$

$$\mathbf{S}(k) = \boldsymbol{\nabla}^T\mathbf{h}\mathbf{C}(k) + \mathbf{R}(k) \quad (5)$$

$$\mathbf{W}(k) = \mathbf{C}(k)\, \mathbf{S}^{-1}(k) \quad (6)$$

$$\mathbf{X}(k) = \mathbf{I} - \mathbf{W}(k)\, \boldsymbol{\nabla}\mathbf{h} \quad (7)$$

$$\boldsymbol{\nu}(k) = \mathbf{z}(k) - \mathbf{h}\left[\hat{\mathbf{x}}(k \mid k-1), \mathbf{u}(k-1)\right].$$

Given that the prediction is consistent and $\mathbf{R}(k)$ is a conservative estimate of the process noise, the update produced by the filter is guaranteed to be consistent [11]. Therefore, assuming a conservative estimate for the observation noise can be determined[1], the problem becomes one of determining a consistent covariance prediction.

When the process noise is independent, the prediction covariance is given by (2). However the process noise is rarely independent because almost any system model contains modeling errors. These modeling errors manifest themselves as time correlations in the process noise [12]. To ensure that the filter remains consistent, the usual approach is to tune (inflate) $\mathbf{Q}(k)$ until the filter becomes consistent. However, if the structure of the process noise model does not approximate the structure of the time correlations very accurately, substantial performance penalties can be accrued [12]. Another approach is to investigate the use of robust prediction algorithms.

[1]For the tracking systems described here this is, in practice, a nontrivial problem because the accuracy of a tracker is often depended on the current configuration and location of a tracking sensor. For this paper we used the manufacturer's specifications.

C. Split Covariance Addition

Suppose that the error in the estimate, $\tilde{\mathbf{x}}(k-1 \mid k-1)$, can be decomposed into two terms

$$\tilde{\mathbf{x}}(k \mid k) = \tilde{\mathbf{x}}_I(k \mid k) + \tilde{\mathbf{x}}_C(k \mid k).$$

The first term corresponds to errors which are known to be independent. The second corresponds to errors which contain time correlations. In the case of head tracking, the first error component corresponds to the integrated effects of observation noises and independent process noise terms. The second component accounts for the correlated process noise terms.

The covariance of $\hat{\mathbf{x}}(k \mid k)$ is

$$\mathbf{P}(k \mid k) = \mathbf{P}_C(k \mid k) + \mathbf{P}_I(k \mid k).$$

Assuming that the process noise can similarly be decomposed into independent and correlated terms

$$\mathbf{v}(k) = \mathbf{v}_I(k) + \mathbf{v}_C(k)$$

with covariances $\mathbf{Q}_I(k)$ and $\mathbf{Q}_C(k)$ respectively, the Split Covariance Addition (SCA) algorithm is

$$\mathbf{P}_I(k \mid k-1) = \mathbf{\Phi}_F(k)\, \mathbf{P}_I(k-1 \mid k-1)\, \mathbf{\Phi}_F^T(k) + \mathbf{Q}_I(k)$$

$$\mathbf{P}_C(k \mid k-1) = \mathbf{\Phi}_F(k)\, \mathbf{P}_C(k-1 \mid k-1)\, \mathbf{\Phi}_F^T(k)/\omega + \mathbf{Q}_C(k)/(1-\omega)$$

where the parameter $\omega \in [0,1]$ is chosen to minimize some measure of uncertainty in $\mathbf{P}(k \mid k-1)$. In [8] it was proved that if the prior estimate is consistent

$$\mathbf{P}(k-1 \mid k-1) - \mathrm{E}\left[\tilde{\mathbf{x}}(k-1 \mid k-1)\, \tilde{\mathbf{x}}^T(k-1 \mid k-1)\right] \geq \mathbf{0}$$

$$\mathbf{P}_C(k-1 \mid k-1) - \mathrm{E}\left[\tilde{\mathbf{x}}_C(k-1 \mid k-1)\, \tilde{\mathbf{x}}_C^T(k-1 \mid k-1)\right] \geq \mathbf{0}$$

and the process noise is consistent

$$\mathbf{Q}_I(k) - \mathrm{E}\left[\mathbf{v}_I(k)\, \mathbf{v}_I^T(k)\right] \geq \mathbf{0}$$

$$\mathbf{Q}_C(k) - \mathrm{E}\left[\mathbf{v}_C(k)\, \mathbf{v}_C^T(k)\right] \geq \mathbf{0}$$

then the estimate

$$\mathbf{P}(k \mid k-1) - \mathrm{E}\left[\tilde{\mathbf{x}}(k \mid k-1)\, \tilde{\mathbf{x}}^T(k \mid k-1)\right] \geq \mathbf{0}$$

$$\mathbf{P}_C(k \mid k-1) - \mathrm{E}\left[\tilde{\mathbf{x}}_C(k \mid k-1)\, \tilde{\mathbf{x}}_C^T(k \mid k-1)\right] \geq \mathbf{0}$$

will be consistent irrespective of the correlation between $\tilde{\mathbf{x}}_C(k-1 \mid k-1)$ and $\mathbf{v}_C(k)$.

The update can be decomposed so that correlated and independent terms are maintained. Replacing (4) by

$$\mathbf{C}(k) = \left(\mathbf{P}_I(k \mid k-1) + \mathbf{P}_C(k \mid k-1)\right) \boldsymbol{\nabla}^T\mathbf{h},$$

$\mathbf{S}(k)$ and $\mathbf{W}(k)$ are calculated using (5) and (6). The updated mean is given by the standard EKF update equation (3) and the updated covariances are

$$\mathbf{P}_I(k \mid k) = \mathbf{X}(k)\,\mathbf{P}_I(k-1 \mid k-1)\,\mathbf{X}^T(k) + \mathbf{W}(k)\,\mathbf{R}(k)\,\mathbf{W}^T(k) \quad (8)$$

$$\mathbf{P}_C(k \mid k) = \mathbf{X}(k)\,\mathbf{P}_C(k-1 \mid k-1)\,\mathbf{X}^T(k) \quad (9)$$

Although the robustness of the SCA algorithm has been proved theoretically [8], no experiments have been carried out to examine its performance in a real, practical system. We now explore the impact of these algorithms in an empirical evaluation.

III. Empirical Study

A. Experimental Setup

Fig. 1. The HiBall™Tracking System. The tracker consists of a set of small, high speed cameras. The room is instrumented with a set of precisely surveyed rails. Each rail contains a set of infrared LEDs which are fired in a predetermined sequence.

This study compared the performance of four implementations of the SCA algorithm (described in Subsection III-C) against an EKF. The algorithms were tested using data sets that were originally acquired for an empirical study, conducted by the University of North Carolina, into the effectiveness of latency and stress in a virtual environment [13]. Users were required to wear an opaque head mounted display (they could only see a computer generated world) and were asked to stand at what was apparently a long drop. As users became more stressed by the environment, this affected the nature of their head movements. The data was collected by the HiBall Tracking System™, whose components are shown in Figure 1. The system is composed of a set of small, high speed cameras which are aligned in a self contained housing and are affixed to the user's head [14]. A set of LEDs are placed into the environment and their positions are surveyed. The LEDs are flashed at high speed in a specific known sequence and, providing the camera can see enough LEDs, the position and orientation of the tracker can be calculated directly through triangulation.

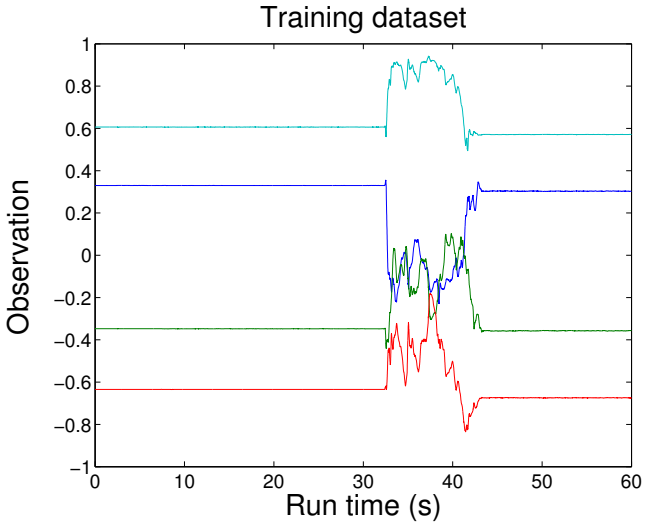


Fig. 2. The training set recorded from the tracker. The tracker returns an average of 160 quaternion measurements per second.

A typical data set collected from this tracker is shown in Fig. 2. This figure shows the nature of head motion: long periods of time in which no activity occurred, followed by rapid and sudden bursts of activity.

B. Filter Implementations

The problem is to estimate the orientation of a user's head in a virtual or augmented reality environment. Because this motion is unconstrained and is in 3D, orientation is represented using quaternions [15]. Because the system needs to predict head orientation several time steps into the future, the angular velocity of the user's head is to be estimated as well. Therefore, the state space consists of

$$\mathbf{x}(k) = \begin{bmatrix} q_x & q_y & q_z & q_w & \omega_x & \omega_y & \omega_z \end{bmatrix}^T \quad (10)$$

where (q_x, q_y, q_z, q_w) is the quaternion and $(\omega_x, \omega_y, \omega_z)$ are the body-fixed angular results.

Assuming the angular velocity is constant and process noise is injected into the angular acceleration, the nominal continuous time process model is

$$\dot{\mathbf{x}}(k) = \mathbf{F}(k)\,\mathbf{x}(k) + \mathbf{G}(k)\,\mathbf{v}(k) \quad (11)$$

where

$$\mathbf{F}(k) = \frac{1}{2}\begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x & q_w & -q_z & q_y \\ \omega_z & 0 & \omega_x & \omega_y & q_z & q_w & -q_x \\ \omega_y & -\omega_x & 0 & \omega_z & -q_y & q_x & q_w \\ -\omega_x & -\omega_y & -\omega_z & 0 & -q_x & -q_y & -q_z \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (12)$$

Implementation	Independent terms	Metric
1	No	Determinant
2	Yes	Determinant
3	No	Trace
4	Yes	Trace

TABLE I
THE CANDIDATE IMPLEMENTATIONS OF SCA WHICH WERE STUDIED.

$$\mathbf{G}(k) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (13)$$

$\mathbf{v}(k) = (v_x, v_y, v_z)^T$ are the angular acceleration noise terms. These equations were integrated using a fourth order Runge-Kutta numerical scheme. Although it is likely that the noise terms will be different about each axis, for this study we assumed that all noise terms were the same and so

$$\mathbf{P}_{vv}(k) = \phi \mathbf{I}_3$$

where ϕ is a positive scalar and $\mathbf{I}_3$ is the 3×3 identity matrix.

The HiBall returns a direct measurement of the orientation quaternion at an update rate of about 160Hz. Therefore, the observation model is linear and is of the form

$$\mathbf{z}(k) = \begin{bmatrix} q_x \\ q_y \\ q_z \\ q_w \end{bmatrix}.$$

From product specifications and studying the logged data from the still behavior of the tracker, the observation covariance was estimated to be

$$\mathbf{R}(k) = \phi_R \mathbf{I}_4$$

where $\phi_R = 9.5 \times 10^{-6}$.

After the update, the quaternion part of the estimate was renormalised by scaling it by the reciprocal of its norm.

C. The Algorithms Studied

Four different implementations of the SCA algorithm were implemented. These are listed in Table I.

The first parameter which was varied was the measure of $\mathbf{P}(k \mid k-1)$ which was minimized. Two measures were studied — the trace and the determinant. The reason why this parameter was varied was that the computational cost for each varies significantly. The value of ω which minimizes the trace is

$$\omega = \frac{\sqrt{A}}{\sqrt{A} + \sqrt{B}},$$

where

$$A = \mathrm{trace}\left[\mathbf{\Phi}_F(k)\, \mathbf{P}_C(k-1 \mid k-1)\, \mathbf{\Phi}_F^T(k)\right]$$

and

$$B = \mathrm{trace}\left[\mathbf{Q}_C(k)\right].$$

However, no closed form solution exists to calculate the value of ω which minimizes the determinant, and numerical schemes must be used instead. Because the SCA equations are convex, numerical schemes such as Newton-Raphson can be used. For the studies presented here we use the `fminbnd` function in Matlab™.

The second parameter which was varied was whether the independent terms were maintained or not. The SCA algorithm propagates two covariance matrices, and this significantly increases the computational costs. If it is assumed that the independent terms are $\mathbf{0}$, only a single covariance matrix needs to be propagated. However, the disadvantage of this approach is that observation noise errors are assumed to be correlated and substantial performance costs might be incurred.

D. Filter Restarting

To provide gating against bad measurements which occasionally arise from the tracker, a normalized innovation test was used. The normalized innovation is defined to be

$$q(k) = \nu^T(k)\, \mathbf{S}^{-1}(k)\, \nu(k). \quad (14)$$

If the innovation is a white, zero-mean Gaussian sequence, the probability distribution of $q(k)$ is a χ^2 distribution whose number of degrees of freedom is equal to the dimensions of $\nu(k)$ [10]. Even if the distribution of $q(k)$ is not χ^2 distributed, the value of the mean provides an important guide to the performance of the filter. If the value is substantially greater than the mean, it implies that the filter is being over confident: the estimate on the innovation covariance is smaller than the actual covariance. Conversely, if it is much smaller than the mean then the filter is being conservative.

A threshold of 40 was used. If $q(k)$ at timestep k was greater than this value, the measurement was skipped and the estimate was set to be the prediction. If the filter did not gate for 10 successive measurements, the filter was reinitialized. The pose was taken from the current measurement, the angular velocity components were set to 0 and the covariance matrix was set to a diagonal matrix whose entries are $(1, 1, 1, 1, 100, 100, 100)$.

E. Filter Tuning

Because the data was empirically collected, there is no truth data which can be used to tune the filter. Therefore, the filters were tuned against a representative data set and a normalized innovation test was used to validate that the filter is correct [10]. Given the filter restarting scheme described in Subsection III-D, ϕ_C and ϕ_I were adjusted until the mean value of the normalized innovation, calculated over 25s–35s of the training set shown in Fig. 2, was less than 4.02. Table II lists the tuned parameters for the EKF and the four SCA implementations. This table shows that although there

Algorithm	ϕ_C	ϕ_I
EKF	0.0	0.7
SCA_1	0.03	0
SCA_2	0.002	0.6
SCA_3	0.03	0
SCA_4	0.002	0.6

TABLE II

THE TUNED PROCESS NOISE PARAMETERS FOR THE FIVE FILTERS FOR RUN 1. VERY SIMILAR RESULTS (WITHIN 2%) WERE OBSERVED WITH THE OTHER FOUR RUNS.

are substantial differences in the noise values needed for the EKF, the SCA with independent terms and the SCA without independent terms, the choice of the metric (trace or determinant) did not have a significant difference.

IV. RESULTS

All four filters were tested against 5 datasets drawn at random from the complete dataset. Figs. 3 and 4 shows the performance of the EKF in one typical run[2]. Fig. 3 shows the normalized innovation of the gated measurements, while Fig. 4 shows time histories of the state estimates and the time histories of the standard deviations (square root of diagonals on covariance histories). To provide the orientation estimate with some physical meaning, the orientation estimates were converted from quaternions to Euler angles. The graphs only show the yaw and estimates for orientation and ω_x for angular velocity. The results for the other states are similar. It should be noted that between approximately 65s and 95s, the sensor did not see a sufficient number of LEDs to generate a position solution. Therefore, during this time interval no data was available which causes the "jump" in the curves. This type of difficulty is not atypical for sensors which rely on line-of-sight and thus should be included in any study of the performance of filtering algorithms. After about 225s the sensor was held more or less stationary.

The normalized innovation illustrates the nature of head movement: the first 225s indicate there is a significant amount of motion. This is followed by 125s during which the user's head is held more or less stationary. These results are confirmed in Fig. 4 which shows rapid changes in the yaw angle and ω_x. However, the orientation and angular velocity estimates remain fairly constant. (The occasional "spiking" occurs due to the occasional drop out of measurements.)

Table III gives the average standard deviation of the estimate for each filter over run 1. As can be seen, the orientation estimates for each filter are extremely close. This is partially due to the fact that the HiBall measures orientation directly and so this estimate will dominate the covariance term. However, it can be seen that the

[2]On the scale of the graphs, the results from the SCA filters are indistinguishable from those of the EKF. For clarity and space reasons we only show the results for the EKF.

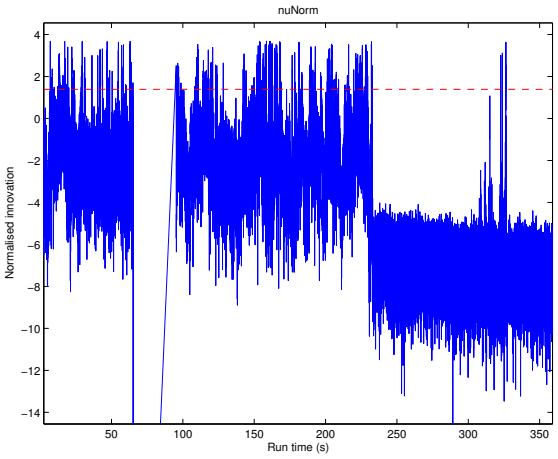
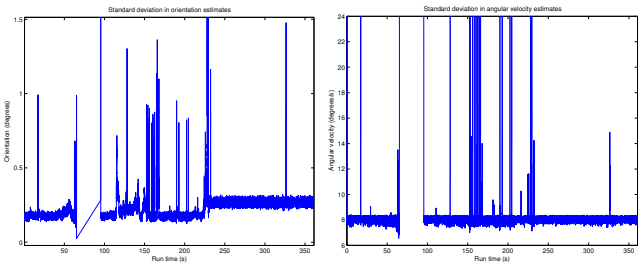
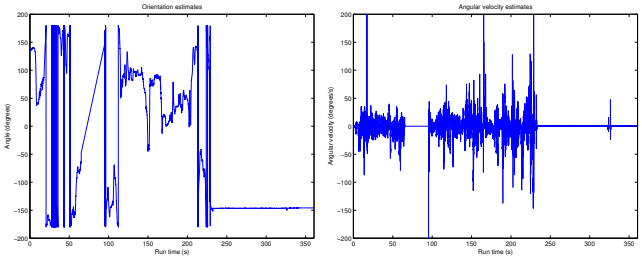


Fig. 3. The log of the normalized innovation of the EKF for run 1. The dashed line shows log(4) which should be the average value if the filter is consistent. The drop between 65s and 95s occurs because the tracker looses track and does not provide any observations.



(a) Covariance history of the orientation estimate.

(b) Covariance history of the angular velocity estimate.



(c) State history of the orientation estimate.

(d) State history of the angular velocity estimate.

Fig. 4. Covariance histories and state estimates for run 1 using the EKF.

	EKF	SCA_1	SCA_2	SCA_3	SCA_4
θ	0.22067	0.22593	0.21729	0.22683	0.21728
ϕ	0.16740	0.18734	0.16804	0.18987	0.17230
ψ	0.22062	0.22587	0.21724	0.22678	0.21723
ω_x	9.9589	8.6395	9.5411	8.5721	9.5404
ω_y	9.9589	8.6395	9.5411	8.5721	9.5404
ω_z	9.9589	8.6395	9.5411	8.5721	9.5404

TABLE III

AVERAGE STANDARD DEVIATION OF ESTIMATES.

	EKF	SCA_1	SCA_2	SCA_3	SCA_4
Run 1	29	20	30	24	30
Run 2	34	33	37	34	37
Run 3	14	13	15	13	15
Run 4	16	15	16	16	16
Run 5	7	7	8	7	8

TABLE IV
FILTER RESTART COUNTS.

EKF has the smallest orientation estimate of all the filters and so has a slightly more accurate overall estimate of orientation. Much greater variation is seen in the angular velocity estimates. The angular velocity standard deviations are the same for all axes. The values for all SCA filters are smaller than those for the EKF. Because the filter is acting consistently, this suggests that the SCA filters are compensating for some unmodelled correlations. The results also suggest that the forms of SCA which do not maintain the independent terms (SCA_1 and SCA_3) perform better than those which do (SCA_2 and SCA_4). One possible explanation is that the observation noise sequence is not truly independent for the HiBall. There are several reasons. First, the HiBall tracker internally maintains its own filtering and estimation algorithms. These access data at a much higher rate (1-2kHz) than the data which is output from the tracker and could be introducing time correlated terms. Furthermore, errors in surveying LED positions can lead to consistent time correlated error terms. The results also show that SCA_2, which is computationally the cheapest implementation of SCA (do not propagate independent terms; minimize trace using the closed form equation) also has the best performance.

The robustness of the SCA implementations was explored by examining their restart counts. Table IV lists the restart counts for all the different filters across all the runs. Apart from SCA_4, all the other SCA algorithms restart less frequently than the EKF. Furthermore, SCA_1 consistently restarts the least frequently.

These results suggest that SCA_1 is the most successful form of the SCA algorithm for the experimental scenario. It restarts less frequently than the EKF and has an angular velocity estimate whose variance is, on average, smaller than that of the EKF. However, these improvements in performance are tempered by a slight degradation in orientation accuracy.

V. CONCLUSION

This paper studied the problem of developing filters for human body tracking and, in particular, the problem of estimating the orientation of a head. We examined the performance of four implementations of a robust prediction algorithm (Split Covariance Intersection) compared with a Kalman Filter. The results suggest that one implementation of SCA can provide a more robust filter which estimates angular velocity more accurately than an EKF. However, this improvement comes at the cost of a slightly worse orientation estimate.

Although these results suggest that the advantages of SCA are marginal, we believe that they are sufficiently encouraging to warrant further study. For example, we plan to investigate how SCA affects the performance of a bank of filters.

ACKNOWLEDGMENTS

We would like to thank Sharif Razzaque and the University of North Carolina at Chapel Hill for making their datasets available for use in this study.

REFERENCES

[1] G. Welch and E. Foxlin, "Motion tracking: No silver bullet, but a respectable arsenal," *IEEE Computer Graphics and Applications*, vol. 22, no. 6, pp. 24–38, 2002.

[2] D. Bowman, E. Kruijff, J. LaViola, and I. Poupyrev, "An Introduction to 3D User Interface Design," *Presence: Teleoperators and Virtual Environments*, vol. 10, pp. 96–108, January 2001.

[3] J. LaViola, "A discussion of cybersickness in virtual environments," *ACM SIGCHI Bulletin*, vol. 32, no. 1, pp. 47–56, 1999.

[4] C. Wareand R. Balakrishnan, "Reaching for objects in vr displays: Lag and frame rate," *ACM Transactions of Computer-Human Interaction*, vol. 1, no. 4, pp. 331–356, 1994.

[5] R. Azuma, *Predictive Tracking for Augmented Reality*. PhD thesis, UNC Chapel Hill, 1995.

[6] D. W. Kyger and P. S. Maybeck, "Reducing Lag in Virtual Displays Using Multiple Model Adaptive Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 1237–1247, 1998.

[7] L. Chai, K. Nguyen, B. Hoff and T. Vincent, "An adaptive estimator for registration in augmented reality," in *Proc. of IEEE International Workshop on Augmented Reality*, (San Francisco, CA, UA), pp. 23–32, IEEE, IEEE Press, October 1999.

[8] S. J. Julier and J. K. Uhlmann, "General Decentralized Data Fusion With Covariance Intersection (CI)," in *Handbook of Data Fusion* (D. Hall and J. Llinas, ed.), (Boca Raton FL, USA), CRC Press, 2001.

[9] J. Liang,C. Shaw, and M. Green, "On Temporal-Spatial Realism in the Virtual Reality Environment," in *Proceedings of User Interface Software and Technology'91*, pp. 19–25, 1991.

[10] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. New York NY, USA: Academic Press, 1988.

[11] T. Nishimura, "On the a priori information in sequential estimation problems," *IEEE Transactions on Automatic Control*, vol. TAC-11, pp. 197–204, April 1966.

[12] S. J. Julier and H. F. Durrant-Whyte, "On the Role of Process Models in Autonomous Land Vehicle Navigation Systems," *IEEE Transactions on Robotics and Automation*, vol. 19, pp. 1–14, February 2003.

[13] M. Meehan, F. Brooks Jr., S. Razzaque, and M. Whitton, "Effect of latency on presence in stressful environments," in *Proceedings of the 2003 IEEE Virtual Reality Conference*, pp. 141–148, IEEE Press, 2003.

[14] G. Welch, G, Bishop, L. Vicci, S. Brumback, K. Keller, and D. Colucci, "High-performance wide-area optical tracking: The hiball tracking system," *Presence: Teleoperators and Virtual Environments*, vol. 10, pp. 1–21, January 2001.

[15] K. Shoemake, "Animating rotations with quaternion curves," in *Proceedings of SIGGRAPH 85*, pp. 245–254, ACM Press, 1985.