Comparison of Orientation Filter Algorithms for Realtime Wireless Inertial Posture Tracking

A. D. Young
Institute for Computing Systems Architecture
School of Informatics, University of Edinburgh
10 Crichton Street Edinburgh EH8 9AB, United Kingdom
Email: ayoung9@inf.ed.ac.uk

***Abstract*—Advances in the miniaturisation of inertial sensors have allowed the design of compact wireless inertial orientation trackers. Such devices require data fusion algorithms to process sensor data into estimated orientations. This paper examines the problem of inertial sensor data fusion and compares two alternative methods for orientation estimation: complementary filtering and Kalman filtering. Experiments are presented to assess the performance and accuracy of the resulting filters. The complementary filter structure is demonstrated to require up to nine times less execution time, while maintaining better accuracy across different movement scenarios, than the Kalman filter structure.**

I. Introduction

Wireless inertial posture tracking offers the possibility to perform realtime capture of a subject's movements over an almost unlimited area. In contrast to traditional optical, ultrasonic, or magnetic capture systems, inertial tracking requires no external infrastructure to define the tracking volume. This capability in turn allows for the development of new applications in diverse areas, such as assisted living, computer interaction and animation.

Many inertial measurement units have been developed within the Wireless Sensor Network (WSN) community. These vary from simple activity detection devices based on accelerometers [1], [2] to full 3-degree of freedom (3DoF) orientation tracking devices comprising accelerometers, magnetometers and rate gyroscopes [3]–[5].

By combining data from multiple 3DoF trackers mounted on the major limb segments of a subject's body, with a forward kinematic model of the subject's skeleton a three-dimensional representation of the subject can be recreated [6].

In order to reconstruct the major parts of the body fifteen sensors are required [7]. Existing commercial solutions [8], [9] use a wired network on the body with a central wireless transmitter. This imposes restrictions on the subject as they must wear a special suit to mount the sensors and cables which may reduce freedom of motion. Fully wireless systems reduce the encumbrance of the subject as sensors are fully autonomous and can be worn in discrete straps. An example wireless posture capture system in operation, with a simple ten sensor body model, is shown in Fig. 1

Fully wireless posture tracking presents challenges in how to stream data from multiple sensor devices in realtime. Transmitting raw sensor data, at the high update rates required for accurate numerical integration of rate gyroscope data, requires a relatively high data rate per device. Systems using this technique are severely limited in the number of devices they can support on a single radio channel [10] and cannot support full body tracking on a single channel. An alternative approach is to process the sensor data locally on each device and transmit the estimated orientation at a lower frequency. As numerical integration is performed on the device the transmitted update rate can be substantially reduced while still conforming to the Nyquist sampling criterion. In previous work [11] this approach has been demonstrated to provide up to a 79% reduction in data rate compared to transmitting raw data. By utilising local processing to reduce transmitted data rate the Orient system is capable of tracking the full posture of a subject, at a frame rate of 64Hz, using a single 250kbps radio channel.

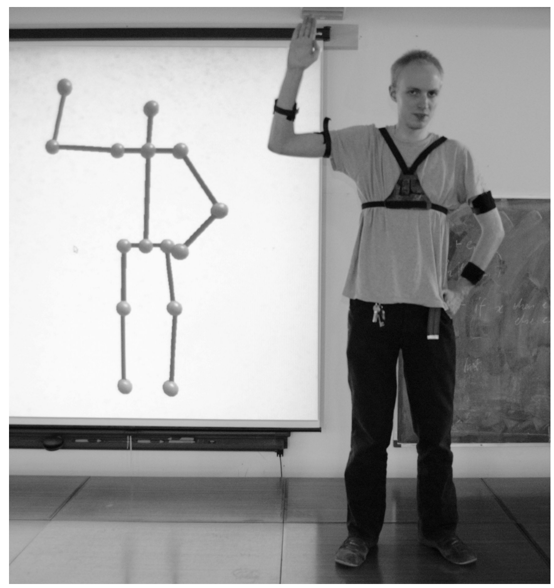

Fig. 1. Orient wireless posture capture system in action

In order to perform local estimation of device orientation it is necessary to design an orientation estimation filter suitable for use on a low power WSN device. Such devices, powered by small batteries, require low complexity algorithms in order to

978-0-7695-3644-6/09 $25.00 © 2009 IEEE
DOI 10.1109/BSN.2009.25

reduce power consumption and processing latency. This paper examines two orientation estimation algorithms, their accuracy and their processing requirements. Section II provides an overview of orientation estimation theory, while Section III presents an experimental comparison of the different filter implementations.

II. Theory

The goal of the orientation estimation filter is to provide an estimate of the rotation between the local body co-ordinate frame of the sensor and a global fixed world frame. 3DoF inertial measurement devices typically use nine sensors comprising triads of accelerometers, magnetometers and rate gyroscopes. These allow the device to measure the components of the device acceleration vector, magnetic field vector and rotational rate vector in the local body co-ordinate frame.

Orientation estimation is typically performed by fusing estimates from two separate estimation methods: rate gyroscope integration and vector observation.

Rate gyroscope integration provides an estimate of the relative rotation from an initial known rotation. As the angular velocity measured by the rate gyroscopes is directly integrated this method provides smooth estimates even during rapid movement.

$$\hat{q}_t = \hat{q}_{t-1} + \frac{1}{2\Delta t}(0, \vec{\omega}) \otimes \hat{q}_{t-1} \tag{1}$$

Rate gyroscope integration can be implemented very efficiently using the difference equation (1) where $\hat{q}$ is the estimated orientation, in quaternion form; Δt is the sample period; $\vec{\omega}$ is the angular rate vector in radians per second; and $\otimes$ is the quaternion multiplication operator. After each update the estimated quaternion should be re-normalised to minimise the effects of rounding errors in limited precision implementations.

The integration process has two significant disadvantages. Firstly, any bias in $\vec{\omega}$ will result in an increasing cumulative error in the estimated orientation. Secondly, the initial orientation of the device must be known.

Vector observation provides an estimate of the orientation relative to a fixed world co-ordinate frame. By measuring the position of two, or more, vectors in the local co-ordinate frame of a device and comparing these with the known position of the vectors in the fixed co-ordinate frame the rotation between the two frames can be calculated. Formally, we want to find the rotation R such that

$$\vec{b}_i = R\vec{w}_i \quad \forall i \in (1, \ldots, n) \tag{2}$$

where $\vec{b}_1, \ldots, \vec{b}_n$ are the set of observed vectors in the local body co-ordinate frame, and $\vec{w}_1, \ldots, \vec{w}_n$ are the set of reference vectors in the global world co-ordinate frame. For wireless inertial orientation trackers the reference vectors used are the direction of acceleration due to gravity, defining the z-axis of the world co-ordinate frame, and the direction of the Earth's magnetic field vector projected into the horizontal plane, defining the world x-axis.

In general, with vector observations corrupted by noise, no solution for R exists. Non-optimal solutions solve this by discarding some of the information contained in one of the vectors. For example, the algorithm used on the Orient devices is defined as

$$\vec{e}_3 = \frac{\vec{a}}{\|\vec{a}\|} \tag{3a}$$

$$\vec{e}_1 = \frac{\vec{m} - \vec{e}_3(\vec{e}_3 \cdot \vec{m})}{\|\vec{m}_t - \vec{e}_3(\vec{e}_3 \cdot \vec{m}_t)\|} \tag{3b}$$

$$\vec{e}_2 = \vec{e}_3 \times \vec{e}_1 \tag{3c}$$

$$\hat{R} = [\vec{e}_1 : \vec{e}_2 : \vec{e}_3]^T \equiv \hat{q} \tag{3d}$$

where $\vec{a}$ and $\vec{m}$ are the observed acceleration and magnetic vectors respectively. This method discards the vertical component of the magnetic field measurement. A similar approach is taken by the TRIAD [12] and FQA [13] algorithms. Optimal solutions, such as the QUEST [14] algorithm, minimise Wahba's [15] loss function

$$L(R) = \frac{1}{2}\sum_{i=1}^{n} a_i|\vec{b}_i - R\vec{w}_i|^2 \tag{4}$$

where a_i are non-negative weights applied to each vector to account for their individual accuracy.

Vector observation has the advantage that it provides an absolute estimate of orientation, however, as observations of the acceleration due to gravity are corrupted by acceleration due to subject movement, it suffers from high frequency noise.

In order to provide accurate orientation estimates, which preserve the high frequency response of rate gyroscope integration and the absolute estimate provided by vector observation, data fusion must be performed. Two approaches will be discussed: complementary filtering and Kalman filtering.

A. Complementary Filtering

Complementary filters can be used to combine two different measurements of a common signal with different noise properties to produce a single output. A typical example would be combining measurements of a signal with low and high frequency noise. By choosing complementary low– and high–pass filters the resulting transfer function applied to the signal is

$$H(s) = H_{LP}(s) + H_{HP}(s) = 1 \tag{5}$$

This has the advantage that no group delay is applied to the signal.

The two estimation approaches previously outlined are suitable for integration into a complementary filter. Recall that, the rate gyroscope integration method suffers from low frequency drift, while the vector observation method suffers from high frequency movement errors.

We can now define a complementary filter equation

$$\hat{q}_t = \begin{cases} q'_t + \frac{1}{k}(q''_t - q'_t) & |\|a\| - 1| < a_T \\ q'_t & |\|a\| - 1| \geq a_T \end{cases} \tag{6}$$

where q' and q'' are the rate gyroscope integration and vector observation estimates respectively, k is a filter co-efficient that controls blending of the two estimates, and a_T is a threshold for optionally gating the vector observation during linear accelerations. High frequency response is maintained by the first term while low frequency drift correction is provided by the second term. The filter co-efficient, k, controls the rate at which drift correction is performed. For small values of k drift correction is fast but more movement induced noise is passed through the filter while larger values result in less high frequency noise but slower drift correction.

In order to reduce movement noise further a gating process is used where drift correction is only performed when the magnitude of the measured acceleration vector is within a bound of $1g$. This limits the number of erroneous drift corrections by only performing corrections while the device is not experiencing significant linear accelerations.

B. Kalman Filtering

The Kalman filter structure is often used for orientation estimation [3], [16]–[18]. Kalman filters use knowledge of the expected dynamics of a system to predicate future system states given the current state and a set of control inputs. Formally this is given by

$$\vec{x}_{k+1} = A\vec{x}_k + B\vec{u}_k + \vec{w}_k \tag{7}$$

where: $\vec{x}$ is the state vector, A is a state transition matrix relating the previous state to the next state, $\vec{u}$ is a control input vector, B is a matrix relating control inputs to system states and $\vec{w}$ is a process noise vector with covariance matrix Q_k representing uncertainty in the system dynamics.

Now given a set of indirect measurements of the state

$$\vec{y}_k = C\vec{x}_k + \vec{v}_k \tag{8}$$

where C is an observation matrix relating system state to observed measurements and $\vec{v}$ is a measurement noise vector with covariance matrix R_k, the Kalman filter can be defined as [19]

$$K_k = AP_kC^T(CP_kC^T + R_k)^{-1} \tag{9a}$$

$$\hat{\vec{x}}_{k+1} = (A\hat{\vec{x}}_k + B\vec{u}_k) + K_k(\vec{y}_{k+1} - C\hat{\vec{x}}_k) \tag{9b}$$

$$P_{k+1} = AP_kA^T + Q_k + AP_kC^TR_k^{-1}CP_kA^T \tag{9c}$$

where, K is the Kalman gain, and P is the estimation error covariance matrix. The Kalman filter can be thought of a predictor-corrector filter with the first term of (9b) the prediction, according to the process model, and the second term the correction, based on indirect measurements of the system state.

An intuitive understanding of the filter operation can be gathered by examining the Kalman gain equation(9a).

$$\lim_{R_k \to 0} K_k = AC^{-1} \qquad \lim_{P_k \to 0} K_k = 0 \tag{10}$$

Substituting the limits of (10) into 9b, we can see that as the measurement error, R_k, decreases greater weight is given to the correction, while as the estimation error, P_k, decreases greater weight is given to the prediction. The Kalman filter can be shown to be the optimal recursive filter in the sense that it minimises the estimated error covariance. The Kalman filter is defined for linear systems, however, it can be extended to support non-linear systems by linearising about the current state. The resulting Extended Kalman filter (EKF) is no longer strictly optimal [20].

For the purpose of this paper an EKF will be simulated using a process model, specifically designed for realtime posture tracking, proposed by Yun *et al* [16]. The state vector is composed of the rotational rate vector and the estimated orientation quaternion, while the measurement vector comprises the observed angular rates and an estimated quaternion derived from a vector observation technique. The angular velocity of a limb segment is modelled by coloured noise generated by a linear system with a white noise input. Two parameters, the noise variance and the time constant of the linear system can be used to tune the behaviour of the filter.

III. Experimental Comparison

In order to compare the behaviour of different data fusion methods they were simulated using Python filter implementations fed by data captured by a real device. Both the complementary filter and the EKF make use of vector observation techniques so these were implemented to a common interface to allow them to be interchanged. The complementary filter structure was simulated using the Orient, TRIAD, FQA and QUEST algorithms, while the EKF was simulated using the QUEST algorithm, as originally proposed by Yun *et al.*, and with the Orient algorithm.

In all experiments sensor data, corrected for offset and scale errors, were collected from a single Orient inertial measurement unit, at its native sample rate of 256Hz, and stored to disk. The different filter implementations were then simulated using the captured data. For the complementary filters the filter co-efficient, k, was set to 128 and an accelerometer gating value, a_T, of $0.1g$ was used to reduce gross movement noise. These values were chosen empirically as they produced reasonable results. For the Kalman filter process model a deviation of $50rad^2/s^2$ was used with a time constant of $0.5s$ [21].

A. Performance

The performance of each of the filter implementations was determined by measuring the time taken to process 1000 samples gathered by a device resting in a static orientation. To minimise the effects of processor load in a pre-emptive operating system, the experiment was repeated ten times and the minimum time taken for each implementation. The results of the performance experiment, normalised to the speed of the fastest implementation, are displayed in Fig. 2. The fastest implementations are the complementary filters using non-optimal vector observations methods. The non-optimal methods are approximately 2.5 times faster than the complementary filter using the optimal QUEST algorithm, and 7-9 times faster than the Kalman filter implementations.

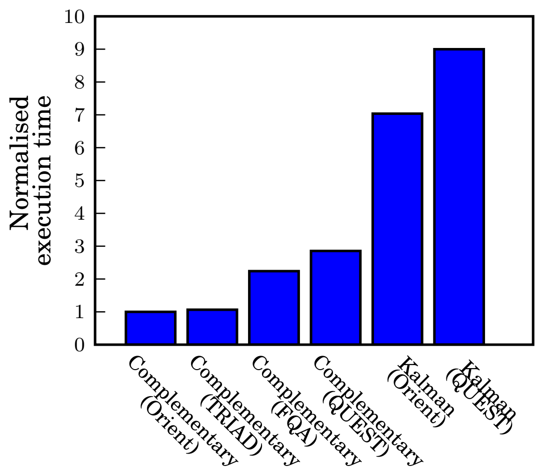

Fig. 2. Comparison of execution time to process 1000 samples

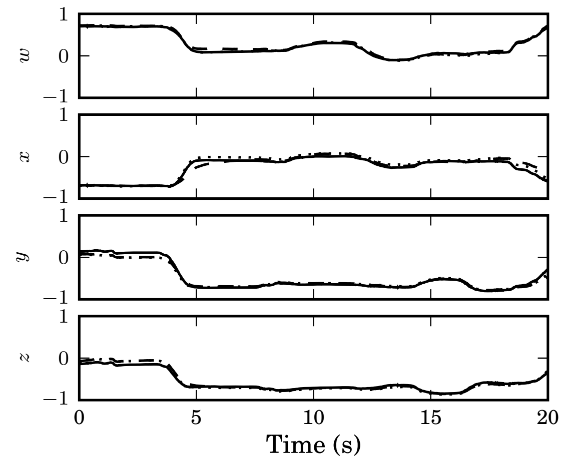

Fig. 3. Comparison of estimated quaternion components for simple motion. Optical – solid, Complementary – dotted, Kalman – dashed

B. Accuracy

In order to provide an accurate truth reference for accuracy comparisons the Orient sensor was mounted on a rigid frame with three optical markers. The positions of these markers were captured using a Qualisys optical motion capture system synchronised to sample at the same instant as the device. This optical data was used to calculate the true orientation of the device using the QUEST vector observation algorithm. In order to assess the accuracy of each filter implementation the rotation between the truth reference and the filter estimate was calculated, factorised into the standard aerospace Euler angle sequence, and the Root Mean Square (RMS) and peak errors calculated over all samples.

Two different movement scenarios were tested:

1) Simple motion – the device was hand-held and gently moved between orientations. A single twenty second capture was performed.
2) Walking motion – the device was mounted on the lower leg and the subject walked at a normal pace on a treadmill. Five trials were performed, each lasting thirty seconds, with a single subject.

In both scenarios the device was initially at rest and movement commenced after approximately three seconds. The initial rest period allows the filter estimate to converge on the true orientation before the motion begins.

For the simple motion scenario all filters tracked the motion reasonably accurately, as illustrated by Fig. 3. Examination of the resulting errors, shown in Table I, reveals that the complementary filters have lower errors in general than the Kalman filters. The optimal QUEST vector observation method performs almost identically to the non-optimal methods. It can be seen that the majority of the error is found in the yaw component which in all filters is mainly determined by the measured magnetic field vector. The Earth's magnetic field can easily be distorted by ferrous objects in the vicinity of the sensor. The lab in which the experiment was conducted presented particular difficulty due to the presence of steel force measurement plates in the floor of the optical capture volume.

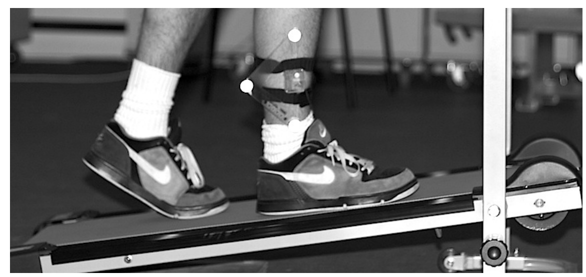

Fig. 4. Experimental setup for capturing walking motion data

In order to test the filter responses to different movement scenarios the comparison experiment was repeated with a device attached to the lower leg of a subject walking on a treadmill, as shown in Fig. 4. The walking scenario presents a greater challenge for the estimation algorithms as the device is in constant motion and also experiences significant shocks as the foot hits the ground. As can be seen from Fig. 5 the complementary filters maintain a reasonable track of the orientation quaternion. However, the Kalman filters perform poorly. The factorised accuracy results for all filter implementations, for the first trial, are shown in Table II. Similar results were produced for the remaining trials. The complementary filters are clearly more accurate than the Kalman filter. The increase in yaw error for all implementations is attributed to the proximity of the sensor to the steel frame of the treadmill.

C. Discussion

The results of the simulation demonstrate that the complementary filter uses significantly less processing power than the Kalman filter. Using the optimal QUEST vector observation method results in minimal improvement in orientation estimation accuracy, however, processing requirements more than double.

TABLE I
FACTORISED ESTIMATION ERROR ANGLES – SIMPLE MOTION

Filter	RMS Roll Error (Peak)	RMS Pitch Error (Peak)	RMS Yaw Error (Peak)
Complementary (Orient)	3.15° (6.68°)	1.87° (3.54°)	8.98° (17.89°)
Complementary (TRIAD)	3.15° (6.68°)	1.87° (3.54°)	8.98° (17.89°)
Complementary (FQA)	3.14° (6.65°)	1.87° (3.54°)	8.98° (17.89°)
Complementary (QUEST)	3.23° (6.34°)	1.26° (3.41°)	9.51° (19.20°)
Kalman (Orient)	3.95° (11.52°)	5.42° (16.58°)	9.58° (25.31°)
Kalman (QUEST)	4.10° (10.84°)	5.18° (16.88°)	10.12° (25.08°)

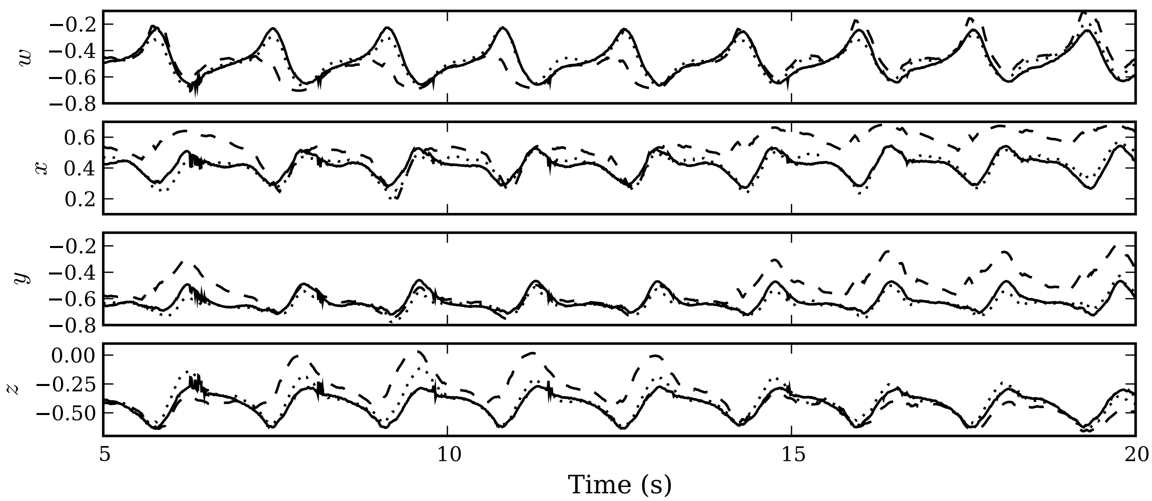

Fig. 5. Comparison of estimated quaternion components for walking motion. Optical – solid, Complementary – dotted, Kalman – dashed

TABLE II
FACTORISED ESTIMATION ERROR ANGLES – WALKING MOTION

Filter	RMS Roll Error (Peak)	RMS Pitch Error (Peak)	RMS Yaw Error (Peak)
Complementary (Orient)	3.16° (10.17°)	3.20° (13.60°)	10.78° (22.29°)
Complementary (TRIAD)	3.16° (10.17°)	3.20° (13.60°)	10.78° (22.29°)
Complementary (FQA)	3.16° (10.17°)	3.20° (13.60°)	10.78° (22.29°)
Complementary (QUEST)	4.00° (11.83°)	2.47° (7.22°)	10.88° (26.49°)
Kalman (Orient)	10.04° (30.74°)	8.46° (25.12°)	31.81° (65.93°)
Kalman (QUEST)	10.13° (31.96°)	11.86° (27.77°)	30.79° (65.98°)

The complementary filter can be implemented using only basic vector and quaternion maths operations allowing efficient implementation of the filter on embedded processors supporting Multiply Accumulate (MAC) operations. A fixed point version of the filter has been implemented on a Microchip dsPIC30F3014 processor. The dsPIC is designed for digital signal processing applications and supports single-cycle MAC operations. The embedded implementation of the filter requires only 280μs to compute a complete update with the processor clocked at 7.37MHz.

The Kalman filter requires numerous matrix operations including multiplies and inversions. The signal processing libraries for the dsPIC require that matrix inversion is performed using floating point maths. Inverting a 7×7 matrix, simulated using the Microchip dsPIC simulator, requires 1.3ms to compute. At least one such inversion is required per update.

The poor accuracy performance of the Kalman filter in the walking test is not altogether surprising. The process model used in the simulation assumes that the angular velocity of a device can be modelled as a coloured noise signal. This model, while reasonable for the gesturing motions it was designed for, is not a good approximation of the dynamics of the leg during walking. The inaccurate process model causes the Kalman filter to generate bad estimates which must be corrected. Alternative process models, designed specifically for the motion of the lower leg while walking, could potentially provide an increase in accuracy over the complementary filter.

The dependency on an accurate process model is a major disadvantage of using Kalman filters for human posture tracking. The human musculoskeletal system is extremely versatile and capable of many different motions. Different parts of the body have different dynamics and therefore process models must be specialised for individual body parts and movement scenarios. Designing Kalman filters for posture tracking is made even harder by the fact that the control inputs to the system, the nerve impulses from the brain controlling the muscles, are hidden from the filter.

IV. Conclusions

Two sensor data fusion filter structures have been discussed. Kalman filtering, commonly chosen because of its history in control theory, does not provide a good solution to the problem of human posture tracking. Unlike traditional Kalman filter problems, such as aircraft attitude estimation, the process model and control inputs are difficult or impossible to estimate. Furthermore, different parts of the human body may require different process models or parameter values. The assumption of a process model governing the motion of a body part causes the Kalman filter to produce incorrect estimates when the process model is inaccurate. Complementary filtering, as it has no assumptions of process dynamics, does not suffer from these problems.

Complementary filters, due to their low complexity, require significantly less processing resources than Kalman filters. The complementary orientation filter outlined in this paper has been simulated to perform up to nine times faster than an EKF specifically designed for realtime posture estimation. The use of non-optimal vector observation methods minimise processing costs while introducing minimal loss of estimation accuracy. Lower processing requirements directly result in lower energy consumption as the processor can spend longer periods in low power sleep states.

Acknowledgements

This research was supported by grants from the Scottish Funding Council (R37329) and the UK Science and Engineering Research Council (C523881) for the Research Consortium in Speckled Computing.

References

[1] C. Park, J. Liu, and P. Chou, "Eco: an ultra-compact low-power wireless sensor node for real-time motion monitoring," *Information Processing in Sensor Networks, Fourth International Symposium on*, pp. 398–403, 2005.

[2] E. Farella, L. Benini, B. Riccò, and A. Acquaviva, "MOCA: A low-power, low-cost motion capture system based on integrated accelerometers," in *Advances in Multimedia*. Hindawi Publishing Corporation, 2007.

[3] T. Harada, H. Uchino, T. Mori, and T. Sato, "Portable orientation estimation device based on accelerometers, magnetometers and gyroscope sensors for sensor network," in *Multisensor Fusion and Integration for Intelligent Systems, Proceedings of IEEE International Conference on*, 2003, pp. 191–196.

[4] A. Lynch, B. Majeed, J. Barton, F. Murphy, K. Delaney, and S. O'Mathuna, "A wireless inertial measurement system (WIMS) for an interactive dance environment," *Journal of Physics: Conference Series*, vol. 15, pp. 95–100, 2005.

[5] V. van Acht, E. Bongers, N. Lambert, and R. Verberne, "Miniature wireless inertial sensor for measuring human motions," in *Engineering in Medicine and Biology Society, Proceedings of the 29th Annual International Conference of the IEEE on*, Aug. 2007, pp. 6278–6281.

[6] E. R. Bachmann, "Inertial and magnetic tracking of limb segment orientation for inserting humans into synthetic environments," Ph.D. dissertation, Naval Postgraduate School, Monterey, California, 2000.

[7] R. Zhu and Z. Zhou, "A real-time articulated human motion tracking using tri-axis inertial/magnetic sensors package," in *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 12, no. 2, June 2004, pp. 295–302.

[8] *Moven Datasheet*, Xsens Motion Technologies. [Online]. Available: http://www.xsens.com

[9] *IGS-190-M*, Animazoo. [Online]. Available: http://www.animazoo.com/IGS190.aspx

[10] *InertiaCube3 Datasheet*, InterSense, Inc. [Online]. Available: http://www.intersense.com

[11] A. Young, M. Ling, and D. K. Arvind, "Orient-2: A realtime wireless posture tracking system using local orientation estimation," in *Embedded networked sensors, Proceedings of the 4th workshop on*. New York, NY, USA: ACM, 2007, pp. 53–57.

[12] G. Lerner, *Spacecraft Attitude Determination and Control*. Kluwer Academic Pub, 1978, ch. Three-Axis Attitude Determination, pp. 420–428.

[13] Y. Xiaoping, E. Bachmann, and R. McGhee, "A simplified quaternion-based algorithm for orientation estimation from Earth gravity and magnetic field measurements," *Instrumentation and Measurement, IEEE Transactions on*, vol. 57, no. 3, pp. 638–650, March 2008.

[14] M. Shuster and S. Oh, "Three-axis attitude determination from vector observations," *Journal of Guidance and Control*, vol. 4, no. 1, pp. 70–77, 1981.

[15] G. Wahba, "A least squares estimate of satellite attitude," *SIAM Review*, vol. 7, no. 3, pp. 384–386, 1965.

[16] X. Yun and E. R. Bachmann, "Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking," *Robotics, IEEE Transactions on*, vol. 22, no. 6, pp. 1216–1227, 2006.

[17] J. Torres, B. O'Flynn, P. Angove, F. Murphy, and C. O. Mathuna, "Motion tracking algorithms for inertial measurement," in *Body area networks, Proceedings of the ICST 2nd international conference on*. ICST, Brussels, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 1–8.

[18] E. Foxlin, "Inertial head-tracker sensor fusion by a complementary separate-bias Kalman filter," in *Virtual Reality Annual International Symposium, Proceedings of the IEEE*, 1996, pp. 185–194, 267.

[19] D. Simon, "Kalman Filtering," *Embedded Systems Programming*, June 2001. [Online]. Available: http://academic.csuohio.edu/simond/courses/eec644/kalman.pdf

[20] G. Welch and G. Bishop, "An introduction to the Kalman filter," *ACM SIGGRAPH 2001 Course Notes*, 2001.

[21] X. Yun, C. Aparicio, E. Bachmann, and R. McGhee, "Implementation and experimental results of a quaternion-based Kalman filter for human body motion tracking," *Robotics and Automation, Proceedings of the 2005 IEEE International Conference on*, pp. 317–322, April 2005.