

# INTEGRATED NAVIGATION OF CAMERAS FOR AUGMENTED REALITY

Thomas B. Schön and Fredrik Gustafsson

Division of Automatic Control, Linköping University,  
SE-581 83 Linköping, Sweden,  
e-mail: (schon, fredrik)@isy.liu.se

**Abstract:** In augmented reality, the position and orientation of the camera must be estimated very accurately. This paper will propose a filtering approach, similar to integrated navigation in aircraft, which is based on inertial measurements as primary sensor on which dead-reckoning can be based, and features in the image as supporting information to stabilize the dead-reckoning. The image features are considered to be sensor signals in a Kalman filter framework. *Copyright©2005 IFAC.*

**Keywords:** Sensor fusion, Kalman filter, Inertial navigation, Augmented reality, Computer vision, Feature extraction

## 1. INTRODUCTION

The idea in *augmented reality (AR)* is to add synthetic background and objects to streaming video images in real-time, while allowing the camera to move. One of the major technical challenges to achieve this is to determine the camera's position and orientation in 3D with very high accuracy and low latency. Typical applications of such a system includes studio recordings with synthetic scenes (Thomas *et al.*, 1997) and virtual reconstruction of historical buildings (Vlahakis *et al.*, 2002).

Prior work in this recent research area focuses on image processing algorithms, where the streaming image is the primary information source (Davison, 2003). This requires quite a lot of features in each image, and has led to a development of marker-based systems, where bar-coded markers are installed in the studio (Thomas *et al.*, 1997). Later work has tried to avoid artificial markers, by including other information like accelerations and angular velocities from inertial sensors (You *et al.*, 1999; You and Neumann, 2001).

When it comes to using vision in AR two fundamentally different strategies have been used:

- The environment is prepared in advance using artificial markers, which impose a significant additional cost to these systems. Examples of this kind of system can be found in (Caarls *et al.*, 2003; Yokokohji *et al.*, 2000; Thomas *et al.*, 1997).
- Markerless systems, which use natural features occurring in the real scene as markers. The approach presented in this paper will utilize this strategy. The need for this kind of systems is motivated in (Azuma *et al.*, 1999). A common characteristic of these systems is that they use some kind of model of the scene. Some attempts to create such a system are given in (You *et al.*, 1999; Klein and Drummond, 2003).

In this contribution, the reverse approach is applied. An inertial measurement unit (IMU) with three degrees of freedom accelerometers and gyroscopes is used as the primary source of information. Dead-reckoning gives a position and orientation relative to the initial camera location. This estimate will quite soon drift away and become completely useless, unless it is supported with secondary sensors, which in this case are provided by the images.

Our approach mimics the navigation systems in aircraft (Nordlund, 2002; Schön *et al.*, 2004). There are obviously many similarities of aircraft navigation and our approach to augmented reality: the aircraft and camera have the same state vector, navigation is based on dead-reckoning IMU sensor signals, and both have to be supported by secondary information. For aircraft, infrastructure based positions from instrument landing systems or satellite positioning systems can be used. In military applications terrain navigation systems can be employed (Bergman *et al.*, 1999). In this paper, features in the image are used as secondary sensors in two different ways:

- Feature displacement: An observed movement of a distinct feature in the image can be directly related to a movement in the camera, which will be shown to correspond to a one-dimensional measurement equation for each feature displacement.
- Recognition of known 3D-objects: Certain characteristic features in the scene are stored in a scene model prior to filtering. When such a feature is observed in the image, two degrees of freedom of the camera position can be determined.

A possible third direction is to use the homography (Hartley and Zisserman, 2003). However, this is not elaborated on within this article. This idea has previously been discussed in e.g., (Diel *et al.*, 2005; Vidal *et al.*, 2001).

By using the IMU as primary sensor, it is not necessary that all six degrees of freedom are present in the features in every image. This is the main advantage in the approach of fusing information from the inertial sensors with the information from the vision sensor.

## 2. DYNAMIC MOTION MODEL

The dynamic state equations for the camera consist of a set of nonlinear differential equations describing how the camera pose is related to the readings from the accelerometers and the gyroscopes according to

$$\dot{x}(t) = f(x(t), u(t), t), \quad (1)$$

where the state vector  $x(t)$  consists of position  $c_f$ , velocity  $v_f$  ( $v_f = \dot{c}_f$ ) and orientation (represented using unit quaternions)  $q$  ( $q = q_{cf}$ ), i.e.,  $x = [c_f^T, v_f^T, q^T]^T$ , where

$$c_f = \begin{bmatrix} c_x \\ c_y \\ c_z \end{bmatrix} \quad v_f = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \quad q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (2)$$

Regarding the notation,  $c_f$  is used to describe the position of the camera center (point  $C$  in Fig. 1) expressed in the  $F$ -system. Furthermore, the accelerometer,  $a_c$ , and the gyroscope,  $\omega_c$ , readings are considered to be input variables,  $u$ , i.e.,

$$u = \begin{bmatrix} a_c \\ \omega_c \end{bmatrix} \quad (3)$$

In the subsequent sections the relevant coordinate systems are defined and the nonlinear function  $f(\cdot)$  in (1) will be derived.

### 2.1 Geometry and Coordinate Systems

The following three coordinate systems are used:

- (1) **Fixed (F):** This is considered to be an inertial system (the rotation of the earth is ignored), which is fixed to earth. The navigation will be performed in this system. Furthermore, the scene model is given in this system.
- (2) **Camera (C):** This coordinate system is attached to the camera and hence it is moving with the camera. Its origin is located in the camera center.
- (3) **Image (I):** The image is projected into this coordinate system, which is located at the principal point.

These three coordinate systems are illustrated in Fig. 1. Furthermore, a fourth coordinate system, the

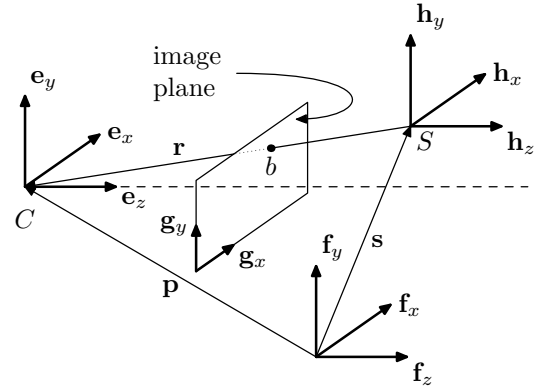


Fig. 1. Illustration of the different coordinate systems and how they are related. Point  $C$  is the position of the camera (optical center) and point  $S$  is the position of a certain static feature in the real scene.

sensor system, is used. This is the coordinate system in which the inertial measurements are obtained. It is not discussed in this paper, which implies that a somewhat unrealistic assumption is used, namely that the inertial sensors are placed in the camera center. However, everything discussed in this paper can rather straightforwardly be adapted to the fact that the sensor coordinate system is present as well.

### 2.2 Position

The position of the camera is given by the position of the camera center (point  $C$  in Fig. 1). The accelerometers measures the inertial forces w.r.t. an inertial system (the  $F$ -system in this work). Hence, the accelerometers will measure the difference between

the acceleration of the camera,  $a_f$ , and the gravity vector,  $g_f$ . However, since the accelerometers are attached to the camera (strapdown inertial system) the measurements will be resolved in the camera coordinate system, according to

$$a_c = R_{cf}(a_f - g_f), \quad (4)$$

where  $R_{cf}$  is a rotation matrix which rotates vectors from the  $F$ -system to the  $C$ -system. Notice that the accelerometer measurement can be modelled as a measurement signal (Rehbinder and Hu, 2004),  $y = a_c$ , or as an input signal,  $u = a_c$ , (common in the aircraft industry). In this work the accelerometer signal is modeled as an input signal, in order to avoid additional states. However, by including the acceleration and the angular velocity in the state vector the acceleration and angular velocity can be modeled by shaping the process noises for these states. The dynamic motion model is according to Newton's second law a double integration of the measured acceleration:

$$\dot{c}_f = v_f, \quad (5a)$$

$$\dot{v}_f = R_{fc}a_c + g_f. \quad (5b)$$

By assuming that the input signal is piecewise constant it is straightforward to derive a discrete-time version of (5).

### 2.3 Orientation

Finding a suitable representation for the orientation of a rigid body in 3D is a more intricate problem than one might first guess. In Section 2.2 rotation matrices (commonly referred to as Direction Cosine Matrices (DCM)) were used to describe rotations. These matrices belong to a group called  $SO(3)$ , defined by

$$SO(3) = \{R \in \mathbb{R}^{3 \times 3} : RR^T = I, \det R = +1\}. \quad (6)$$

The name  $SO$  stands for *special, orthogonal*, due to the constraints (6) (Murray *et al.*, 1994). Hence, the most natural description to use is DCM. However, this description has some problems, since it requires six parameters and since it is hard to enforce the orthogonality condition. It has been shown that five is the minimum number of parameters that have to be used in order to parameterize the rotation group in such a way that a global description, without singular points is obtained (Hopf, 1940; Stuelpnagel, 1964). However, the dynamics for this parameterization is quite complicated, which implies that it is not used. Using four parameters, unit quaternions<sup>1</sup>, to describe the orientation provides the best alternative, since it is a representation that is nonsingular and the dynamics is linear (bilinear if the angular velocity is modelled as a state variable) in the states. The downside is that the unit constraint has to be maintained and that the parameterization is non-global. However, this non-global property will not be a problem in practice.

Another commonly used parameterization is the Euler angles. The advantage of this parameterization is that it only requires three parameters, but the dynamics is nonlinear and it is a singular, non-global representation. According to the authors the best trade-off for parameterizing the rotation group is provided by the unit quaternion. Hence, all computations are performed using unit quaternions. However, when the orientation is presented to the user Euler angles are used, since this parameterization is easiest to interpret and visualize.

A good account of the twelve most common rotation parameterizations is given in (Shuster, 1993). Furthermore, (Shoemaker, 1985) provides some good intuition regarding the unit quaternions. The dynamic equation for the quaternions is

$$\dot{q}(t) = \frac{1}{2}\Omega(\omega)q(t), \quad (7)$$

where

$$\Omega(\omega) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \quad (8)$$

The quaternion has to be normalized, i.e.,

$$q^T(t)q(t) = 1, \quad (9)$$

in order to represent an orientation. By invoking the assumption that the angular velocity is constant between the sampling instants the rotation vector  $\theta$  can be defined as

$$\theta = \omega_t T_s, \quad (10)$$

and under this assumption it can be shown that the solution to (7) is

$$q_{t+1} = A(\theta)q_t, \quad (11)$$

where  $A(\theta)$  can be shown to be

$$A(\theta) = \cos(\|\theta\|/2)I_4 + \frac{\sin(\|\theta\|/2)}{\|\theta\|}\Omega(\theta). \quad (12)$$

Care has to be taken when estimating the orientation, since the set of all rotations,  $SO(3)$  is not a vector space, but rather a manifold, due to the constraint (6). Using quaternions this is handled simply by normalizing the estimate. However, the best would of course be if an estimator could be derived that delivered estimates, which inherently existed on the manifold. The problem is that the resulting problem is nonconvex.

## 3. USING VISION AS A SENSOR

In order to be able to incorporate the information available in the image sequence into the estimation problem measurement equations

$$y_t = h(x_t, e_t, t), \quad (13)$$

have to be derived. These equations should describe the relationship between the state variable,  $x$ , and the information available in the images. In the subsequent sections two different approaches on how to

<sup>1</sup> Another name for the unit quaternion is Euler-Rodrigues symmetric parameters, or Euler symmetric parameters (Shuster, 1993).

derive these equations are discussed. Since a single image contains large amounts of information the most essential information has to be efficiently recovered. The approach using in this work is to extract features from the images. In the computer vision literature an *image feature* is any structural feature that can be extracted from the image. The idea of using inertial sensors and features extracted from the images have previously been exploited e.g., in (Rehbinder and Ghosh, 2003; Jiang *et al.*, 2004).

### 3.1 Camera Model

A camera is a device that provides two dimensional projections of a three dimensional real scene. The camera model describes this projection in mathematical terms. Hence, the camera model is most essential in forming the measurement equations. The camera model used in this work is the *pinhole model* (Hartley and Zisserman, 2003),

$$[x_i, y_i]^T = [fx/z, fy/z]^T, \quad (14)$$

where  $[x_i, y_i]^T$  are the coordinates for the feature in the image coordinate system, see Fig. 1. Furthermore,  $[x, y, z]^T$  is the corresponding position in the real scene and  $f$  is the focal length. The model (14) is simply a way to state that two objects lying on the same ray will be projected onto the same point in the image plane. This model is used due to its simplicity. However, all equations derived in this paper can be extended to more advanced camera models including parameters for optical distortion etc. For more details on different camera models the reader is referred to (Hartley and Zisserman, 2003).

### 3.2 Two Dimensional Feature Displacements

The goal of this section is to derive a measurement equation using the fact that an observed movement of a static feature in the image can be directly related to a movement of the camera. This measurement equation can then be used within the Kalman filtering framework. The derivation starts with the simple fact that,

$$\mathbf{r} = \mathbf{r} \Leftrightarrow 0 = \mathbf{r} + x\mathbf{e}_x + y\mathbf{e}_y + z\mathbf{e}_z, \quad (15)$$

where the  $\mathbf{r}$ -vector, defined in Fig. 1, has been expressed in two different coordinate systems, the rotating  $C$ -system and the fixed feature system. Differentiating (15) w.r.t. time gives

$$0 = \dot{\mathbf{r}} + \dot{x}\mathbf{e}_x + \dot{y}\mathbf{e}_y + \dot{z}\mathbf{e}_z + x\dot{\mathbf{e}}_x + y\dot{\mathbf{e}}_y + z\dot{\mathbf{e}}_z. \quad (16)$$

From Fig. 1;  $\dot{\mathbf{r}} = \dot{\mathbf{p}} - \dot{\mathbf{s}} = \dot{\mathbf{p}}$ , which together with the fact that  $x\dot{\mathbf{e}}_x + y\dot{\mathbf{e}}_y + z\dot{\mathbf{e}}_z = \boldsymbol{\omega} \times \mathbf{r}$  gives

$$\dot{x} = -z\omega_y + y\omega_z - v_x, \quad (17a)$$

$$\dot{y} = -x\omega_z + z\omega_x - v_y, \quad (17b)$$

$$\dot{z} = -y\omega_x + x\omega_y - v_z, \quad (17c)$$

where  $\mathbf{v} = \dot{\mathbf{p}}$ . Differentiating (14) gives

$$\dot{x}_i = f \frac{\dot{x}z - x\dot{z}}{z^2}, \quad \dot{y}_i = f \frac{\dot{y}z - y\dot{z}}{z^2}. \quad (18)$$

Inserting (14) and (17) in (18) gives

$$\dot{x}_i = \underbrace{\frac{x_i y_i}{f} \omega_x - f \left(1 + \frac{x_i^2}{f^2}\right) \omega_y + y_i \omega_z}_{\dot{x}_{i,R}} + \underbrace{\frac{-f v_x + x_i v_z}{z}}_{\dot{x}_{i,T}}, \quad (19a)$$

$$\dot{y}_i = \underbrace{f \left(1 + \frac{y_i^2}{f^2}\right) \omega_x - \frac{x_i y_i}{f} \omega_y - x_i \omega_z}_{\dot{y}_{i,R}} + \underbrace{\frac{-f v_y + y_i v_z}{z}}_{\dot{y}_{i,T}}, \quad (19b)$$

where the velocity has been split into one rotational part (indicated with subscript  $R$ ), and one translational part (indicated with subscript  $T$ ). It is impossible to use (19) to gain perfect information about the present position and orientation of the camera, which has previously been discussed in i.e., (You *et al.*, 1999; Matthies *et al.*, 1988; Longuet-Higgins and Prazdny, 1980). However, in combination with the other sensors these equations will help in the task of finding the position and orientation of the camera. Gyroscopes provide measurements of the angular velocity,  $\boldsymbol{\omega}$ , and hence the rotational terms in (19) can be considered to be known (with a certain degree of uncertainty).

The measurements are the projection of the features in the image plane, i.e.,

$$\mathbf{y}^j = \begin{bmatrix} x_i^j \\ y_i^j \end{bmatrix} + \mathbf{e}_t^j, \quad j = 1, \dots, N, \quad (20)$$

where  $N$  is the number of features, and  $\mathbf{e}$  the measurement noise. However, since (19) is used the measurement equations will be implicit, i.e., the measurement equations will not be in the form (13), but rather in the following form:

$$0 = h(\mathbf{y}_t, \mathbf{y}_{t-1}, \mathbf{x}_t, \mathbf{e}_t, t). \quad (21)$$

There is one problem with the derived measurement equations, the depth information  $z$  of the feature is still present. This problem can be tackled in numerous ways. The first idea that comes to mind is to extend the state vector with the depth states  $z^i, i = 1, \dots, N$ . In (Davison, 2003) an algorithm similar to the particle filter is used to estimate the depths. Alternatively, the depths can be thought of as *nuisance* variables which should be eliminated. Substituting  $z$  from (19a) into (19b) gives

$$\frac{\dot{y}_i - \dot{y}_{i,R}}{\dot{x}_i - \dot{x}_{i,R}} = \frac{-f v_y + y_i v_z}{-f v_x + x_i v_z}, \quad (22)$$

which is the resulting one-dimensional measurement equation. It is straightforward to rewrite (22) on the form (21), using the Euler approximation for the differential operator, according to

$$\alpha_t (-f v_{x,t} + x_{i,t} v_{z,t}) = -f v_{y,t} + x_{i,t} v_{z,t}, \quad (23)$$

where

$$\alpha_t = \left( \frac{y_{i,t} - y_{i,t-1}}{T_s} - \dot{y}_{i,R} \right) / \left( \frac{x_{i,t} - x_{i,t-1}}{T_s} - \dot{x}_{i,R} \right).$$

Finally (23) can be written

$$\underbrace{[-f\alpha_t \ f \ (\alpha_t x_{i,t} - y_{i,t})]}_{C_t} v_f = 0. \quad (24)$$

This is the resulting measurement equation for two dimensional feature displacement.

### 3.3 Three Dimensional Features and Model

The vision system delivers a list of  $N$  feature coordinates in the image plane,  $\{x_i^j, y_i^j\}_{j=1}^N$  and the corresponding positions,  $\{s_j = [s_{j,x}, s_{j,y}, s_{j,z}]\}_{j=1}^N$ , in the real scene. This position is obtained from a three dimensional model of the world in which the camera is moving. This model is generated off-line. Intuitively, this information should provide valuable information for estimating the camera pose. Using (14) and Fig. 1 gives

$$[x_i, y_i]^T = [fr_{c,x}/r_{c,z}, fr_{c,y}/r_{c,z}]^T, \quad (25)$$

where  $r_c$  is the vector from the camera center to the current feature. Fig. 1 also reveals that

$$r_c = R_{cf}(c_f - s_f). \quad (26)$$

The resulting measurement equation is found by using the idea from the previous section, i.e., writing the measurement equation in the implicit form (21). This results in

$$0 = \begin{bmatrix} r_{c,z}x_i - fr_{c,x} \\ r_{c,z}y_i - fr_{c,y} \end{bmatrix} + e, \quad (27)$$

which simply corresponds to multiplying (25) with  $r_{c,z}$ . Similar ideas have been presented in e.g., (Davison, 2003). The difference is that in this work an off-line model of the real scene is used in combination with information from inertial sensors. Hence, the costly procedure of preparing the environment with artificial markers is not necessary. Furthermore, the information from the inertial sensors is obtained at a higher frequency than the vision measurements and will be especially helpful during fast maneuvers.

## 4. ILLUSTRATIONS

In order to evaluate the ideas proposed in this paper a virtual environment, briefly described in this section, is used. More specifically a three dimensional model of a car is used. Fig. 2 provides two images from the video sequence. The car is standing still and the camera is moving around the car in such a way that the camera is always facing the car. Since the true position and orientation of the camera is known, the acceleration and angular velocity can be calculated. Using this the inertial measurements can be simulated, simply by adding the proper noise to the true accelerations and angular velocities. Furthermore, the 3D model of the car provides an image sequence from which features can be extracted. These features will constitute the vision measurements, which will be included in the

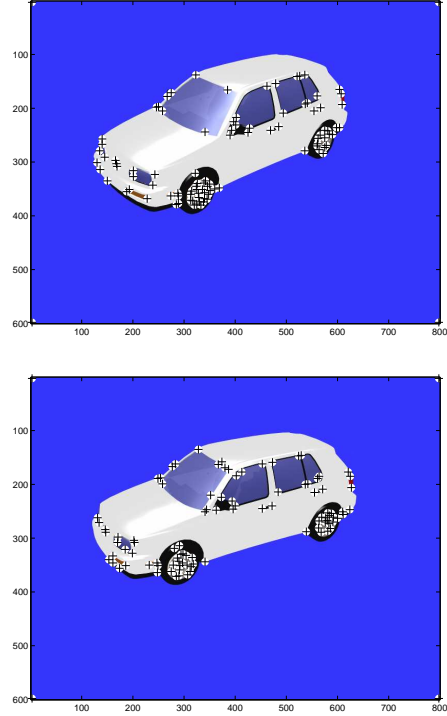


Fig. 2. Two images from the video stream used to obtain the vision measurements. Furthermore, several 2D features have been indicated in the images. The camera has been rotated 10 degrees from the upper to the lower image.

estimation problem using the ideas discussed in Section 3.

The next step is to use authentic inertial and vision measurements, which will be provided by our partners at BBC R&D in London. They have a positioning system installed in their studio (referred to as freeD (Thomas *et al.*, 1997)), providing the true pose, which can be used to assess the estimation performance. The authors are currently working together with Xsens (Xsens, 2005) on using the idea presented in Section 3.3. The preliminary results looks promising.

## 5. CONCLUSIONS

This paper propose a filtering approach for estimating the position and orientation of a camera in three dimensions. The underlying idea of supporting inertial sensors using additional sensors has previously been successfully used for instance within the aircraft industry. The difference is that in this work vision is used, instead of for instance terrain elevation data bases, to support the dead-reckoning of the inertial sensor information. Furthermore, two different strategies regarding the process of incorporating vision measurements in the Kalman filtering framework were discussed. Finally, some brief illustrations on how to evaluate these ideas were given.

## ACKNOWLEDGMENTS

This work has been performed within the MATRIS consortium, which is a sixth framework research program within the European Union (EU), contract number: IST-002013. The authors would like to thank the EU for the financial support and the partners within the consortium for a fruitful collaboration this far. For more information about the MATRIS consortium, please visit the consortium home-page, [www.ist-matris.org](http://www.ist-matris.org).

## REFERENCES

- Azuma, R., J.W. Lee, B. Jiang, J. Park, S. You and U. Neumann (1999). Tracking in unprepared environments for augmented reality systems. *Comp. & Graphics* **23**(6), 787–793.
- Bergman, N., L. Ljung and F. Gustafsson (1999). Terrain navigation using Bayesian statistics. *IEEE Cont. Sys. Mag.* **19**(3), 33–40.
- Caarls, J., P. Jonker and S. Persa (2003). *Lecture Notes in Computer Science*. Chap. Sensor Fusion for Augmented Reality, pp. 160–176. Vol. 2875. Springer Verlag. Berlin.
- Davison, A.J. (2003). Real-time simultaneous localisation and mapping with a single camera. In: *Proc. Ninth IEEE Int. Conf. on Comp. Vis.*. Vol. 2. Nice, France. pp. 1403–1410.
- Diel, D.D., P. DeBitetto and S. Teller (2005). Epipolar constraints for vision-aided inertial navigation. In: *proceedings of the 7th IEEE Workshop on Motion and Video Computing*. Vol. 2. Colorado, USA. pp. 221–228.
- Hartley, R. and A. Zisserman (2003). *Multiple View Geometry in computer vision*. 2 ed.. Cambridge University Press. Cambridge, UK.
- Hopf, H. (1940). Systeme symmetrischer bilinearformen und euklidische modelle der projektiven räume. *Vierteljahrsschrift der Naturf. Gesellschaft in Zürich* **85**, 165–177.
- Jiang, B., U. Neumann and S. You (2004). A robust hybrid tracking system for outdoor augmented reality. In: *Virtual Reality 2004*. Chicago, USA. pp. 3–10.
- Klein, G. and T. Drummond (2003). Robust visual tracking for non-instrumental augmented reality. In: *Int. Symp. on Mixed Augmented Reality*. Tokyo, Japan. pp. 113–123.
- Longuet-Higgins, H.C. and K. Prazdny (1980). The interpretation of a moving retinal image. *Proc. of the Royal Society of London, Series B, Biological Sciences* **208**(1173), 385–397.
- Matthies, L., R. Szeliski and T. Kanade (1988). Kalman filter-based algorithms for estimating depth from image sequences. Technical Report CMU-RI-TR-88-1. The Robotics Institute. Carnegie Mellon University.
- Murray, R.M., Z. Li and S.S. Sastry (1994). *A Mathematical Introduction to Robotic Manipulation*. CRC Press.
- Nordlund, P.-J. (2002). *Sequential Monte Carlo Filters and Integrated Navigation*. Licentiate thesis, Linköping university. Thesis No. 945.
- Rehbinder, H. and B.K. Ghosh (2003). Pose estimation using line-based dynamic vision and inertial sensors. *IEEE Transactions on Automatic Control* **48**(2), 186–199.
- Rehbinder, H. and X. Hu (2004). Drift-free attitude estimation for accelerated rigid bodies. *Automatica* **40**(4), 653–659.
- Schön, T., F. Gustafsson and P.-J. Nordlund (2004). Marginalized particle filters for mixed linear/nonlinear state-space models. *Accepted for publication in IEEE Tran. on Sig. Proc.*
- Shoemake, K. (1985). Animating rotation with quaternion curves. In: *Proc. 12th annual conf. on Comp. graph. and int. tech.*. San Francisco, USA. pp. 245–254.
- Shuster, M.D. (1993). A survey of attitude representations. *The Journal of the Astronautical Sciences* **41**(4), 439–517.
- Stuelpnagel, J. (1964). On the parametrization of the three-dimensional rotation group. *SIAM Review* **6**(4), 422–430.
- Thomas, G.A., J. Jin, T. Niblett and C. Urquhart (1997). A versatile camera position measurement system for virtual reality tv production. In: *International Broadcasting Convention*. Amsterdam. pp. 284–289.
- Vidal, R., Y. Ma, S. Hsu and S. Sastry (2001). Optimal motion estimation from multiview normalized epipolar constraint. In: *8th Int. Conf. on Comp. Vis. (ICCV)*. Vol. 1. Vancouver, Canada. pp. 34–41.
- Vlahakis, V., N. Ioannidis, J. Karigiannis, M. Tsotros, M. Gounaris, D. Stricker, T. Gleue, P. Daehne and L. Almeida (2002). Archeoguide: An augmented reality guide for archaeological sites. *Computer Graphics in Art History and Archaeology* **22**(5), 52–60.
- Xsens (2005). Xsens Technologies, Enschede, The Netherlands. [www.xsens.com](http://www.xsens.com).
- Yokokohji, Y., Y. Sugawara and T. Yoshikawa (2000). Accurate image overlay on video see-through hmds using vision and accelerometers. In: *Virtual Reality 2000 Conference*. New Brunswick, NJ USA. pp. 247–254.
- You, S. and U. Neumann (2001). Fusion of vision and gyro tracking for robust augmented reality registration. In: *IEEE Virtual Reality*. Yokohama, Japan. pp. 71–78.
- You, S., U. Neumann and R. Azuma (1999). Orientation tracking for outdoor augmented reality registration. *IEEE Computer Graphics and Applications* **19**(6), 36–42.