Kalman Filtering of Pose Estimates in Applications of the RAPID Video Rate Tracker

Richard Evans

Roke Manor Research Limited,
Roke Manor Romsey, Hampshire SO51 0ZN

RAPID is a video-rate model based tracker which generates refined estimates of an object's position and orientation (pose) given approximate initial estimates. A practical application of this technique requires (1) prediction from-frame-to-frame of the pose of the object being tracked, to accommodate realistic target movement and (2) temporal integration of pose estimates to reduce measurement noise. These needs are both satisfied by a Kalman filter. In order to apply a Kalman filter, however, we first construct statistical models of both the apparent motion of the object between frames and also of the accuracy of pose measurements made at each processed frame. The filtered tracker output provides a robust estimate of object pose at video rate when implemented in software running on a standard mini-computer. The capabilities of this technique are demonstrated by application to the task of monitoring the pose of an unmanned aircraft during its approach to an airfield and during landing.

The use of a simple video camera to make geometric measurements has much practical appeal in a range of applications, provided of course that viable processing algorithms can be developed. The advantages of vision techniques stem from the low cost of commercially available cameras, the minimal instrumentation requirements and other limitations placed on the design of the sensor platform and the object under investigation. Algorithms have been developed to extract the relative pose of the sensor and viewed object, eg by Lowe [1] and Stephens [2], but these algorithms generally require significant processing time, even when using advanced hardware.

The technique developed by Harris and Stennett [3], which is the basis of the RAPID (Real-time Attitude and Position Determination), is highly efficient and has been demonstrated at video rate with a standard mini-computer. This model-based technique culminates in the determination of pose by minimising (by linear algebra) the quantity

$$E(\mathbf{q})=\sum_i [\ l_i + \mathbf{q}.\mathbf{c}_i\ ]^2,$$

where $\mathbf{q}$ is a 6-vector defining the pose (position and orientation) of the object, l_i is the image plane distance between an observed edge and the predicted position of the i'th edge of the model at a specified control point, and $\mathbf{c}_i$ is a 6-vector whose value depends on the position of the edge within the model and on an initial estimate of the object's pose. (Model edges which pass undetected in the image are simply ignored.)

When applying the RAPID technique to a practical case of a moving object, it is in principle possible to use the pose estimate, calculated by processing one video frame, as the initial estimate of the object's pose in the next video frame. This approach to tracking a moving object has the disadvantage that the object's motion would be limited to small movements between frames since RAPID searches for model edges in a limited region about the predicted position. This problem can be overcome by using a simple predictor, such as an $\alpha-\beta$ tracker [4] which also has the advantage of performing a temporal smoothing of pose estimates, to reduce measurement noise. In practise however, it has been found difficult to set the tracker parameters as the measurement noise depends on the number and position of edges found, and also on the current pose of the object. In some extreme cases, the edges detected in a particular frame may not define all the object's degrees of freedom; clearly a more sophisticated predictor/filter is required.

This paper describes the use of a Kalman filter [5] for pose prediction and filtering. The next section outlines the formulation of a Kalman filter and the following sections describe how the required models are constructed for use in filtering pose estimates. Finally, we demonstrate the technique by applying RAPID to a video sequence recorded in an unmanned aircraft during landing.

KALMAN FILTER OUTLINE

Let $\hat{\mathbf{x}}_t$ be a vector that represents the estimated state of a system at time t. Given a new measurement, $\mathbf{y}_t$, made at that same instant, the state vector

BMVC 1990 doi:10.5244/C.4.16

estimate is updated to $\hat{\mathbf{x}}'_t$, given by

$$\hat{\mathbf{x}}'_t = \hat{\mathbf{x}}_t + K(\mathbf{y}_t - H\hat{\mathbf{x}}_t),$$

where K is the Kalman gain matrix and H is a matrix which maps the estimated state to the corresponding expected observation. Between observations it is assumed that the true state of the system evolves according to

$$\mathbf{x}_{t+1} = A\mathbf{x}_t + \varepsilon_t,$$

where ε_t is a random variable of zero mean and covariance defined by the matrix Q_t. Thus given $\hat{\mathbf{x}}'_t$, $\hat{\mathbf{x}}_{t+1} = A\,\hat{\mathbf{x}}'_t$. If the error in the observation $\mathbf{y}_t$ has zero mean and covariance R_t, and the error in $\hat{\mathbf{x}}_t$ has zero mean and covariance P_t, then the optimal choice of K (that which minimises the trace of P'_t, the covariance of $\hat{\mathbf{x}}'_t$) is

$$K = P_tH^T[HP_tH^T + R_t]^{-1}, \text{ and}$$

$$P'_t = P_t - KHP_t.$$

In the time to the next observation, however, confidence in the state vector estimate worsens because of the uncertainty in evolution, thus

$$P_{t+1} = AP'_tA^T + Q_t.$$

THE OBJECT MOTION MODEL

In this application of Kalman filtering, the RAPID pose estimate, $\mathbf{y}_t$, is the 6-vector pose estimate found by the minimisation of E($\mathbf{q}$). In the simplest moving object case we assume uniform motion, so the state vector contains both position and velocity terms. In particular we write,

$$x = (\mathbf{r}, \bullet, \dot{\mathbf{r}}, \dot{\bullet})^T,$$

where $\mathbf{r}$ is the object's position 3-vector (relative to the sensor), and $\bullet$ is a rotation 3-vector defining its orientation;

$$A = \begin{bmatrix} I_6 I_6 \\ 0_6 I_6 \end{bmatrix} \text{ and}$$

$$H = \left[I_6 0_6\right],$$

where I_6 and 0_6 are the 6-by-6 identity and zero matrices.

We assume that the above motion model is accurate apart from a random fluctuation in velocities, ie

$$Q = \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & Q_1 & Q_2 \\ 0_3 & 0_3 & Q_2^T & Q_3 \end{bmatrix},$$

where Q_1, Q_2 and Q_3 are to be determined.

In many practical situations neither the object being tracked nor the sensor are rigidly fixed, as for example when the object is one aircraft and the sensor is mounted on another flying nearby. In such cases Q will be the combined effect of the motion of the object in global coordinates and the motion of the sensor in global coordinates. In particular, if the sensor rotates, the object will appear to move in sensor-based coordinates and the greater the distance between sensor and object, the greater will be the amount of movement. The following paragraphs outline the computation of Q in terms of the sub-matricies Q_1, Q_2 and Q_3.

Suppose $\mathbf{u}$ and $\mathbf{u}'$ are the velocities of the sensor and object in global coordinates and $\boldsymbol{\omega}$ and $\boldsymbol{\omega}'$ are their rotation-vector rates, and that all the (co)variances of these quantities are known. Thus if $\mathcal{V}$ and $\mathcal{C}$ are the variance and covariance operators, $\mathcal{V}(\omega_1)$ and $\mathcal{C}(\omega_1, u'_2)$ etc, are input parameters for the model. Q_1 is simply the sum of the covariances of $\boldsymbol{\omega}$ and $\boldsymbol{\omega}'$, assuming the motion of the sensor and object are uncorrelated.

Q_2 and Q_3 depend on the current position of the object relative to the sensor. The velocity of the object in sensor coordinates is

$$\mathbf{v} = \boldsymbol{\omega}\times\mathbf{r} - \mathbf{u} + \mathbf{u}',$$

where $\mathbf{r}$ is the position of the object in sensor coordinates. Suppose $\boldsymbol{\omega}$ is subjected to a small fluctuation $\Delta\boldsymbol{\omega}$ about its mean position, and similarly $\boldsymbol{\omega}'$, $\mathbf{u}$, and $\mathbf{u}'$, then the corresponding fluctuation in $\mathbf{v}$ is

$$\Delta\mathbf{v} = \Delta\boldsymbol{\omega}\times\mathbf{r} - \Delta\mathbf{u} + \Delta\mathbf{u}',$$

$$\text{ie} \quad \Delta v_i = \varepsilon_{ijk}\Delta\omega_j r_k - \Delta u_i + \Delta u_i',$$

where ε_{ijk} is the Levi-Cevita symbol and the summation convention is in force. For small fluctuations in the rotation rate of the sensor and object, the corresponding change in relative rotation rate is

$$\Delta\dot{\theta}_i = \Delta\omega_i - \Delta\omega'_i.$$

We can now calculate Q_2, the covariance of the components of object's velocity (in sensor coordinates) with it's components of rotation. Q_2 is

$$\mathcal{C}(\dot{\theta}_i, v_p) =$$

$$\mathcal{E}[(\Delta\omega_i - \Delta\omega'_i)\,(\varepsilon_{pjk}\Delta\omega_j r_k - \Delta u_p + \Delta u_p')].$$

In very many cases, such as aircraft flying nearly parallel, it can be assumed that the motion of the sensor and of the object are uncorrelated ie,

$$\mathcal{E}[\,\Delta\omega_i\,\Delta u_p'] = \mathcal{E}[\,\Delta\omega'_i\Delta\omega_j] = \mathcal{E}[\Delta\omega'_i\Delta u_p] = 0.$$

A further simplification is to assume that the individual components of motion of the sensor and object are uncorrelated with each other, ie $\mathcal{E}[\Delta\omega_i \Delta u_p] = \mathcal{E}[\Delta\omega'_i \Delta u_p'] = 0$ for all i and p, and $\mathcal{E}[\Delta\omega_i \Delta\omega_j] = 0$ for i≠j. In these circumstances the above expression simplifies to

$$Q2=\begin{bmatrix} 0 & -\mathcal{V}(\omega_1)r_3 & \mathcal{V}(\omega_1)r_2 \\ \mathcal{V}(\omega_2)r_3 & 0 & -\mathcal{V}(\omega_2)r_1 \\ -\mathcal{V}(\omega_3)r_2 & \mathcal{V}(\omega_3)r_1 & 0 \end{bmatrix}.$$

(It should be noted that this last assumption is not be wholly accurate for aerodynamic bodies, but it appears to prove adequate for present purposes.)

Similarly Q3, the covariance of the components of the velocity of the object in sensor coordinates is

$$\mathcal{C}(v_i,v_p) = \mathcal{E}[\,(\varepsilon_{ijk}\Delta\omega_j r_k - \Delta u_i + \Delta u_i')\,(\varepsilon_{pmn}\Delta\omega_m r_n - \Delta u_p + \Delta u_p')].$$

With assumptions about sensor and object motion as above, this simplifies to

$$Q3=\begin{bmatrix} r_3^2\mathcal{V}(\omega_2)+r_2^2\mathcal{V}(\omega_3)+\mathcal{V}(u_1)+\mathcal{V}(u'_1) & -r_2r_1\mathcal{V}(\omega_3) & -r_3r_1\mathcal{V}(\omega_2) \\ -r_2r_1\mathcal{V}(\omega_3) & r_1^2\mathcal{V}(\omega_3)+r_3^2\mathcal{V}(\omega_1)+\mathcal{V}(u_2)+\mathcal{V}(u'_2) & -r_3r_2\mathcal{V}(\omega_1) \\ -r_3r_1\mathcal{V}(\omega_2) & -r_3r_2\mathcal{V}(\omega_1) & r_1^2\mathcal{V}(\omega_2)+r_2^2\mathcal{V}(\omega_1)+\mathcal{V}(u_3)+\mathcal{V}(u'_3) \end{bmatrix}.$$

Thus Q_t can be calculated for each filter cycle, from the model input parameters, using the current best estimate of the relative position of the sensor and object.

THE MEASUREMENT MODEL

The confidence we may attach to a particular refined pose estimate depends on how well defined is the minimum of the quadratic surface of function

$$E(\mathbf{q}) = \textstyle\sum_i [\, l_i + \mathbf{q}.\mathbf{c}_i \,]^2 ,$$

where the $\mathbf{c}_i$ depend on the detected feature points at the initially estimated object pose. Imagine an ellipsoid touching the quadratic surface at the minimum such that the second derivatives of the quadratic surface and the ellipsoid match. This ellipsoid is defined by the equation

$$(\mathbf{q}-\mathbf{q}')[\mathbf{c}_i\,\mathbf{c}_i^T](\mathbf{q}-\mathbf{q}')^T = 1,$$

where $\mathbf{q}'$ is the centre of the ellipsoid. We assume that the axes and orientation of this ellipsoid provide a good estimate of the covariance, R, of the resulting measurement. Thus by analogy with normal distribution theory, we set

$$R = \frac{1}{2}\left[\textstyle\sum_i \mathbf{c}_i\,\mathbf{c}_i^T\right]^{-1}.$$

Unfortunately, when only a few control points are detected this inverse cannot be calculated because of ill-conditioning. There are also certain situations when the detected control points do not fully define the pose of the object; in these cases the coefficients matrix is singular. The formula defining the Kalman filter gain can be re-arranged, however, to avoid the need to compute the inverse. Instead of the standard formulation we use

$$K = P_tH^TR_t^{-1}[HP_tH^TR_t^{-1}+I]^{-1},$$

where $R^{-1} = 2\left[\sum_i \mathbf{c}_i\,\mathbf{c}_i^T\right].$

With this formulation for K, the filter gain can be calculated robustly for each filter cycle, weighting each measurement according to its expected accuracy.

DEMONSTRATION

The RAPID algorithms, have been implemented in Pascal running on a VAX3400 machine, hosting an Imaging Technology IT100 card for image capture and display. Although the VAX3400 is a general purpose machine the RAPID algorithms run fast enough for video rate processing to be demonstrated.

Video data has been recorded, for this demonstration, from a forward looking sensor mounted in the nose of an unmanned aircraft. The data has been recorded during approach and landing at an airfield which is modelled visually, as shown in figure 1, by a set of 16 edge features: 4 at each side of the main runway, 2 at each end of the runway, and 2 each side of a section of the perimeter tax-way.

As the aircraft approaches the airfield, under the control of a remote pilot, the video shows the runway growing in the field of view. Figure 2 shows two frames of typical video data, with the airfield model superimposed according to its pose as estimated by RAPID; the white lines mark the edges and ends of the main runway and the selected section of taxi-way. Good alignment between the image data and projected model data can be seen in both images.

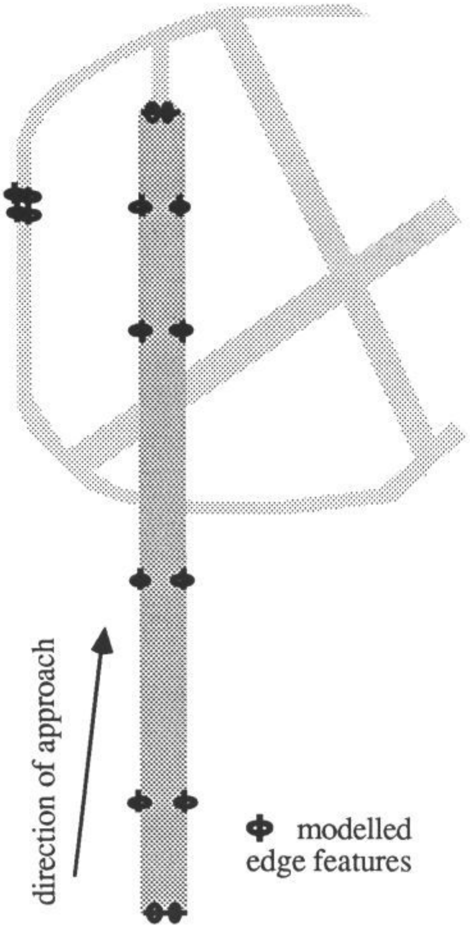

Figure 1 Visual model of runway.

Figures 3 and 4 show a typical track, plotting (in airfield coordinates) the height and cross-runway displacement of the aircraft as a function of down-runway range. The apparently fast initial movement of the aircraft, at about 2200m, is the result of RAPID correcting the very first pose estimate. This acquisition stage is performed manually at present. After acquisition the tracker shows an expected descent path until about 800m from touch down. At this point the airfield model has expanded beyond the camera's field of view so that an insufficient number of features can now be detected to define the model's pose and the tracked path gradually diverges from expectation.

Figures 5, 6 & 7 show the estimated aircraft roll, pitch and yaw. In each figure a set of tracks have been plotted, each obtained by independent processing of the same segment of recorded video data. The differences between these tracks are mainly the result of differences at manual acquisition, in particular differences in the time of start of track. (Some small differences may be the result of analog noise in video-tape replay.) These figures give an indication of the repeatability of the technique, though a more controlled analysis is of course required. The tracks are in close agreement up to the point, at about 800m from touch down, where the model has expanded beyond the camera's field of view. The discrepancies at about 1200m-1400m correspond to a temporary burst of video interference. The tracker is seen to be notably stable despite this period of very noisy data.

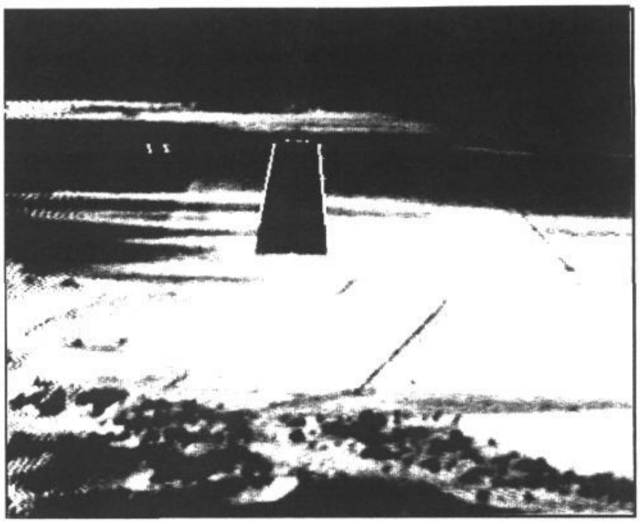

Figure 2a: Typical video frame with projected model outline superimposed.

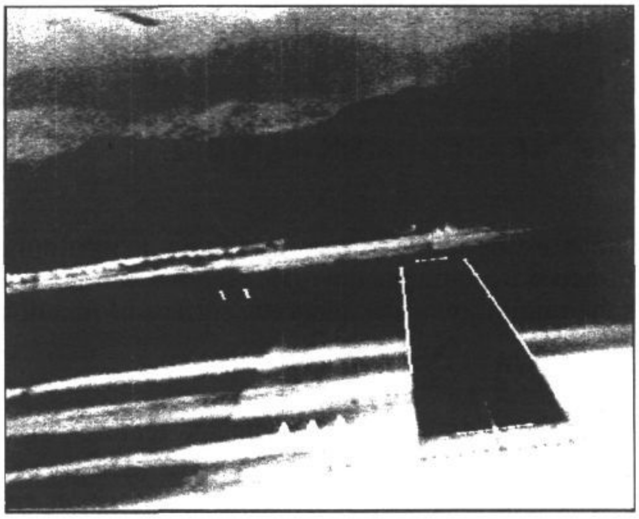

Figure 2b: Typical video frame with projected model outline superimposed.

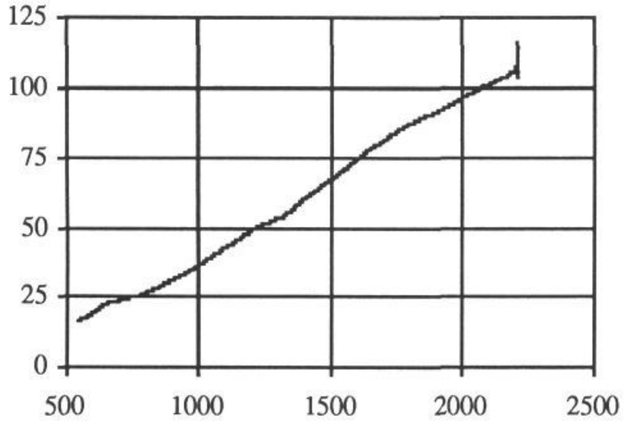

Figure 3. Aircraft height (m) versus down-runway range (m) during landing.

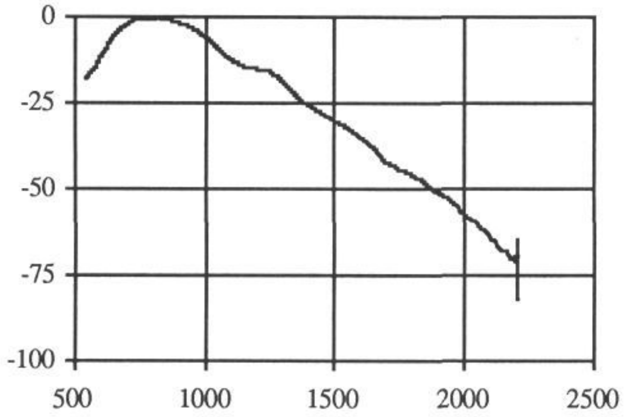

Figure 4. Aircraft cross-runway displacement (m) versus down-runway range (m) during landing.

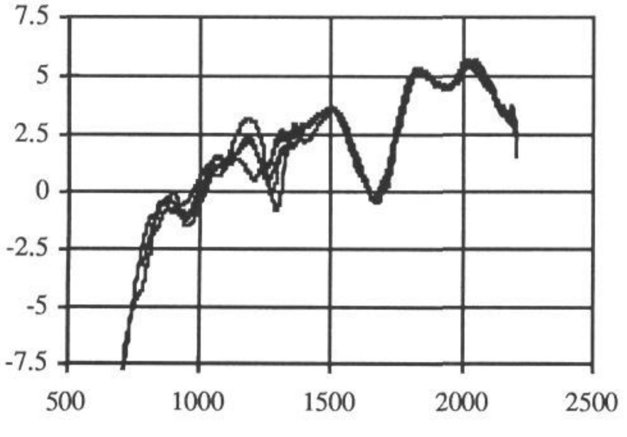

Figure 5. Aircraft roll (degrees) versus down-runway range (m) for five independent passes through video data.

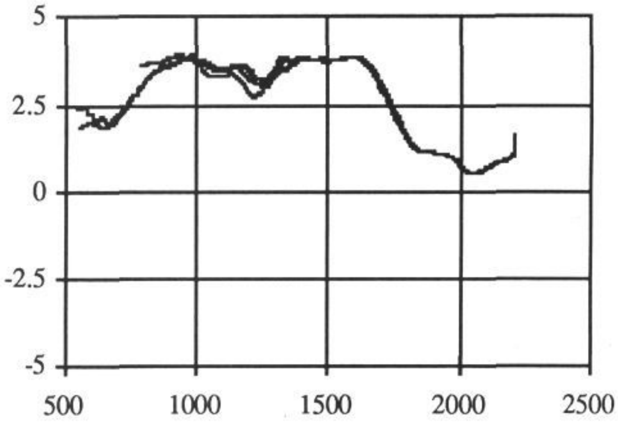

Figure 6. Aircraft pitch (degrees) versus down-runway range (m) for five independent passes through video data.

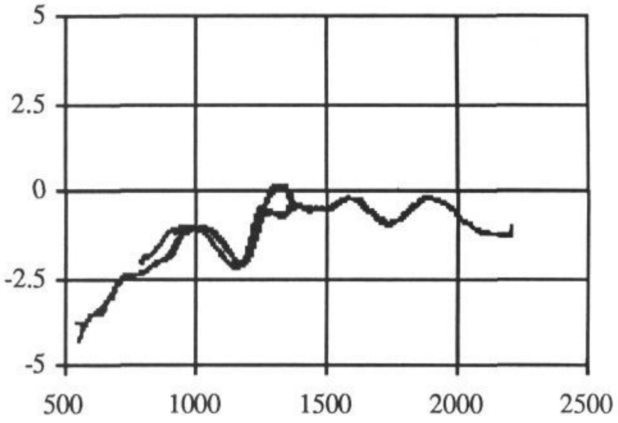

Figure 7. Aircraft yaw (degrees) versus down-runway range (m) for five independent passes through video data.

CONCLUSIONS

The RAPID algorithms combine a fast model-based technique to calculate the pose of a viewed object given an initial estimate, and Kalman filtering for smoothing and pose prediction. These algorithms can be implemented at video rate in a standard mini-computer. They are robust in operation, as has been demonstrated in application to recorded data from a camera mounted in the nose of an unmanned aircraft during landing.

ACKNOWLEDGEMENT

The author thanks Mr Brian Luff of RAE, Farnborough, for his assistance in making available the video data used in this demonstration.

REFERENCES

1 **Lowe, D. G.** "Stabilized Solution for 3D Model Parameters" *Proceedings of the First European Conference on Computer Vision,* ECCV90.

2 **Stephens, R. S.** "Real-Time 3D Object Tracking" *Proceedings of the fifth Alvey Vision Conference,* ACV89.

3 **Harris, C. G. & Stennett, C.** "3D Object Tracking at Video Rate - RAPID" *Proceedings of the first British Machine Vision Conference,* BMVC90.

4 **Blackman, S. S.** *Multiple-Target Tracking with Radar Applications* Artech House Inc, 1986.

5 **Kalman, R. E.** "A new approach to linear filtering and prediction problems" *Trans. ASME, J. of Basic Engineering,* March 1960.