A Fusion and Learning Algorithm for Landing Aircraft Tracking: Compensating for Exhaust Plume Disturbance

Zbigniew Korona

zbigniew@meceng.coe.neu.edu

Mieczyslaw M. Kokar

kokar@meceng.coe.neu.edu

***Abstract*—An algorithm is presented for tracking a landing aircraft using fusion of two different passive sensors, a laser range finder (LRF) and a forward-looking infrared (FLIR) camera. The main feature of this algorithm is its ability to identify and compensate for an exhaust plume disturbance. The algorithm is based on the extended Kalman filter (EKF) and the filtering confidence function (FCF) which introduces a learning approach to the tracking problem. The results of a simulation using the learning tracking algorithm and the EKF alone are presented and compared.**

***Keywords*—fusion, Kalman filtering, neural network, tracking, exhaust plume**

I. Introduction

Passive sensors have recently started to play an important role in target tracking applications [11,12] where radars traditionally have been used. The main advantage of passive sensors is that they do not generate electromagnetic waves and therefore do not clutter the environment with additional signals. Additionally, passive sensors are generally cheaper than active sensors. This work discusses fusion of measurements from two passive sensors for 3-D tracking of a landing aircraft. An infrared ((FLIR) or simply IR) sensor provides the azimuth and elevation angle measurements and a laser range finder (LRF) sensor provides the range measurements. In spite of the fact that the LRF transmits energy, it can be considered a passive sensor in military tracking systems because it utilizes a very concentrated and directed energy beam that makes it very difficult to detect.

The IR sensor provides azimuth and elevation data, but for 3D tracking, range data are also needed. Although LRF provides the necessary range data directly, there have been few attempts to integrate IR and LRF sensors for tracking because they differ significantly in accuracy (high accuracy FLIR vs. low quality range sensor). The issue of fusing signals from two sensors with different accuracy is one of the

Zbigniew Korona and Mieczyslaw Kokar are with the Industrial Engineering Department at Northeastern University, Boston, MA 02115, U.S.A.

This research was partially sponsored by the Center for Electromagnetics Research at Northeastern University, Boston, MA. The authors also wish to thank Ralph Smith from NAWC, Lakehurst, NJ and Stephen Linder from Northeastern University for helpful discussions.

controversies in the sensor fusion literature. If the quality of sensors vary widely, fusion does not improve performance of a tracking system [9] and considerable computational savings can be achieved by eliminating fusion. The IR sensor has its own disadvantages, including susceptibility to the temperature variation of a tracked object. In our application, the aircraft has exhaust plume due to throttle activity required to maintain the glideslope. The plume causes the centroid of the thermal image of the aircraft to move downward in opposition to the ultimate direction of the aircraft's response. This results in a significant error in the elevation angle measured by an IR camera. This error is especially critical for measuring the rate of descent and the rate of departure of the aircraft from the glideslope.

Traditionally, the extended Kalman filter (EKF) is used in tracking applications [3,4]. Unfortunately, in this kind of a problem, where a non-Gaussian noise disturbance caused by exhaust plume is present, a straightforward utilization of the Kalman filter does not give satisfactory results [13]. Therefore we propose an approach that reduces the negative effects of the plume on tracking accuracy. Our solution utilizes a "filtering confidence function" (FCF) that is added to the tracking algorithm. This function implements additional filtering of the measurement data by acting as a prewhitening filter. The strength of this approach strongly depends on the selection of the filtering confidence function which we find by applying a learning algorithm.

Section 2 presents a statement of our tracking problem. Section 3 contains a discussion of three possible approaches to improve the EKF tracking algorithm. Section 4 describes the learning tracking algorithm (LTA). Simulation results are given in Section 5.

II. Tracking Problem

The aircraft dynamics are modeled in Cartesian coordinates as

$$x(k+1) = Ax(k) + v(k), \qquad k = 0, 1, \ldots \tag{1}$$

where $x(k)$ is the 6-dimensional state vector of the tracked object at time k:

$$x = [x \ \dot{x} \ y \ \dot{y} \ z \ \dot{z}]'. \tag{2}$$

A is the transition matrix for the sampling period T:

$$A = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

The variable $v(k)$ is a zero-mean white Gaussian noise with a known variance

$$E[v(k)v(j)'] = Q\delta(k,j), \tag{4}$$

where $\delta(k,j)$ is the Kronecker delta function, and

$$Q = \begin{bmatrix} qT^4/4 & qT^3/2 & 0 & 0 & 0 & 0 \\ qT^3/2 & qT^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & qT^4/4 & qT^3/2 & 0 & 0 \\ 0 & 0 & qT^3/2 & qT^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & qT^4/4 & qT^3/2 \\ 0 & 0 & 0 & 0 & qT^3/2 & qT^2 \end{bmatrix}, \quad (5)$$

where q is the variance of the process noise modeling the acceleration. The state vector includes the position and velocity components of the target in three dimensions, which results in a sixth order tracking filter. This means that a nearly constant velocity motion of the landing aircraft is assumed. This is in accordance with the variable state dimension approach that it is best to track a target using the lowest possible state dimension [2]. It is possible to use only a six order tracking filter for our problem because aircraft velocities were found to be relatively constant enough along the glideslope [10].

The measurements are modeled as

$$z(k) = h[x(k)] + w(k), \quad (6)$$

where $z(k)$ is a three-dimensional vector that consists of range (from LRF), azimuth and elevation angles (from IR), and h is the transformation from Cartesian coordinates to polar coordinates. The transformation for the range is given by

$$r = (x^2 + y^2 + z^2)^{1/2}, \quad (7)$$

for the azimuth angle by

$$a = tan^{-1}(y/x) \quad (8)$$

and for the elevation angle by

$$e = tan^{-1}[z/(x^2 + y^2)^{1/2}]. \quad (9)$$

The vector $w(k)$ represents independent Gaussian measurement noise with zero-mean and known variances such that

$$E[w(k)w(j)'] = R\delta(k, j), \quad (10)$$

where

$$R = \begin{bmatrix} q_r & 0 & 0 \\ 0 & q_a & 0 \\ 0 & 0 & q_e \end{bmatrix} \quad (11)$$

and q_r, q_a, q_e are the variances for the sensor's range, azimuth and elevation measurement noise, respectively.

Tracking an aircraft with this model is normally performed with the extended Kalman filter [3],[4]. The Kalman filter is the optimal linear estimator if both measurement noise and process noise are white, while the extended Kalman filter is used for nonlinear applications. However, the Kalman filter performance degrades rapidly if the assumption of white noise is violated, like in the case of our application.

III. Alternative Tracking Approaches

We have investigated a number of approaches to improve the performance of a landing aircraft tracking system in the presence of additional measurement error caused by plume. The following possibilities were considered:

1. Change the way of processing the measurements [1] , i.e., update the state estimate vector using

$$\begin{aligned} \hat{x}(k+1|k+1) &= \hat{x}(k+1|k) + Gf(z(k+1) \\ &\quad -H(k+1)\hat{x}((k+1|k), s)), \end{aligned}$$

where $\hat{x}(k+1|k+1)$ is the value of the state estimate vector, $\hat{x}(k+1|k)$ is the value of the predicted state vector, and $H(k+1)$ is the Jacobian matrix of h. The main idea here is to generalize the processing scheme by replacing a linear transformation with a more general function f and a gain matrix G. Since both f and G are unknown, they have to be learned by the tracking algorithm. Also, an additional variable s is introduced. It contains some qualitative information relevant to tracking. For example, it tells us whether plume exists or not. However, by giving up the linearity assumption altogether, we lose the advantage of having an optimal filter for the case when the system happens to be linear.

2. Keep the main idea of the Kalman filter, but introduce additional weighting of the state estimate and an appropriate procedure modification for calculating the state estimate error covariance as suggested in [2]. At each scan, the predicted state and its covariance for the IR sensor are first computed [2]. Then, the state estimate $\hat{x}_1(k+1|k+1)$ and its covariance $P_1(k+1|k+1)$ are computed based on the validated IR measurements using the Probabilistic Data Association Filter (PDAF) approach.

   Unfortunately, we cannot directly use the approach as presented in [2] because we do not have enough statistical information about plume. When plume is detected we use a learning approach instead of the PDAF approach. Namely, both the state estimate and its covariance are then adjusted according to:

$$\hat{x}_1(k+1|k+1) = f_1(\hat{x}_1(k+1|k+1), s), \tag{12}$$

   and

$$P_1(k+1|k+1) = g_1(P_1(k+1|k+1), s). \tag{13}$$

   Both functions f_1 and g_1 need to be learned. The s parameter has the same meaning as in the previous approach. Then, the state estimate $\hat{x}(k+1|k+1)$ and its covariance $P(k+1|k+1)$ are calculated based on the validated LRF measurement [2]. The main problem in this approach is to find criteria to learn functions f_1 and g_1.

3. Introduce an additional measurement weighting function and use the extended Kalman filter to process the resulting weighted measurements. In this approach, the influence of plume is reduced by the weighting function and, therefore, does not impact the performance of the extended Kalman filter. The LRF sensor has approximatively the same accuracy during the whole period of tracking. However, the accuracy of measurements originating from the IR sensor is changing. The azimuth measurement accuracy remains the same, but the elevation measurement accuracy depends on whether plume is present (lower accuracy) or not (higher accuracy). It is straightforward to design an adaptive Kalman filter in such a way that the measurement noise covariance matrix changes depending on whether or not plume is present. The function which changes the measurement noise covariance matrix can

be based on the magnitude of plume. A much more important problem is that the measurements, especially the elevation angle, become biased by a constant error when plume appears. In such a situation, the performance of the standard extended Kalman filter, which is designed as a suboptimal filter for nonlinear systems in the presence of white noise, degrades even more. To avoid this degradation a measurement weighting function is introduced.

The list of possible solutions described above is not exhaustive. It contains the directions which we explored during our preliminary analysis. Our conclusion was that the first two solutions are not feasible for our application, for the reasons indicated above. As a result, we have chosen the third direction. The solution introduces a measurement weighting function as a preprocessor to the standard extended Kalman filter, where the function is acquired through the process of neural network learning. The next section presents more details of this approach.

IV. The Learning Tracking Algorithm

The goal of the learning tracking algorithm (LTA) is to reduce the negative effect of plume on tracking accuracy by implementing a prewhitening filter (FCF) to eliminate a constant error (bias) in elevation angle measurements. Classical methods for designing such a filter are not much help because there is not enough statistical information about plume available.

In our approach, the main part of the FCF is implemented by a connectionist (neural) network. It consists of two input nodes (time and speed), twenty nodes with sigmoidal nonlinearities in the first hidden layer, five nodes with sigmoidal nonlinearities in the second hidden layer and one linear node in the output layer. The goal of the FCF is to learn the bias function

$$bias = f(time, velocity).$$

We assume that the influence of plume on the elevation angle measurement error is time invariant for a given aircraft moving with a given speed. This means that the plume influence can be represented as a function of $\Delta t = t - t_0$, where t is time and t_0 is the initial time of the plume appearance (throughout the rest of this paper we take $t_0 = 0$). The velocity input to the FCF comes from the estimate of velocity from the Kalman filter, at the time when plume is detected. The bias may also depend on other factors, like rain or the type of the aircraft. In this study we assumed that the influence of rain is small enough to be omitted from the consideration. Also, we assume that a separate FCF has to be learned for each aircraft type.

The FCF is trained using the standard backpropagation algorithm and data collected by a radar. The accuracy of the radar data is high enough to consider them as the true measurements. The difference between elevation angle measurements originated from the radar and the IR camera was used as the training data for the backpropagation neural network.

Fig. 1 shows the architecture of the LTA. It consists of the following blocks:

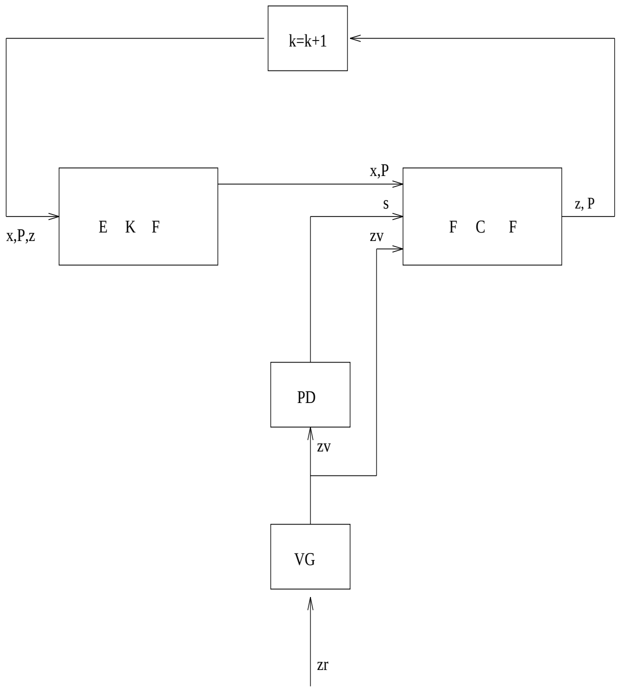


Fig. 1. Architecture of the Learning Tracking Algorithm.

1. EKF is the standard suboptimal extended Kalman filter that gives values of state estimates and their covariances. Inputs are the state estimate vector x, its covariance matrix P, and the measurements vector z.
2. Validation Gate (VG) removes obvious outliers that could degrade the performance of the tracker. In our system we are only concerned with range measurements, because the LRF sensor has a lower accuracy than the FLIR sensor, and because a laser beam could easily miss an aircraft. If the measurement is not validated, then the predicted state is taken as the next state estimate and the predicted state estimate error covariance matrix as the next error covariance matrix. Input is the raw measurements vector z_r.
3. Plume Detector (PD) is a video plume detector of abrupt changes in elevation angle. It is implemented through the use of the Cumulative Sum Algorithm (CSA) [5,6]. If plume is not detected, then only the standard EKF is applied and the FCF is not involved. If plume is detected, then the FCF and EKF are applied in sequence. Input is the validated measurements vector z_v. Output is an indication of the plume appearance s.
4. FCF is a filtering confidence function that acts as a whitening filter. Inputs are the state estimate vector x, its error covariance matrix P, the vector of validated measurements z_v, and the symbolic input (plume) s. Outputs are the filtered measurements z and the modified state error covariance matrix P.

In conclusion, if the plume detector does not detect plume, only the standard extended Kalman filter is used. Also, any outliers caused by other phenomena are not processed by the EKF. They are eliminated by the validation gate. If the plume detector does detect plume, the measurements (in particular the elevation angle measurements) are additionally processed through the learning block FCF, and the state estimate covariance matrix P is changed to accommodate an increase in the state estimation error caused by the appearance of plume.

V. Results of Experiments

We have tested our algorithm on simulated data. The plume was modeled as the product of velocity and the arctangent function of time in the presence of additive Gaussian noise. This form of the function was selected so as

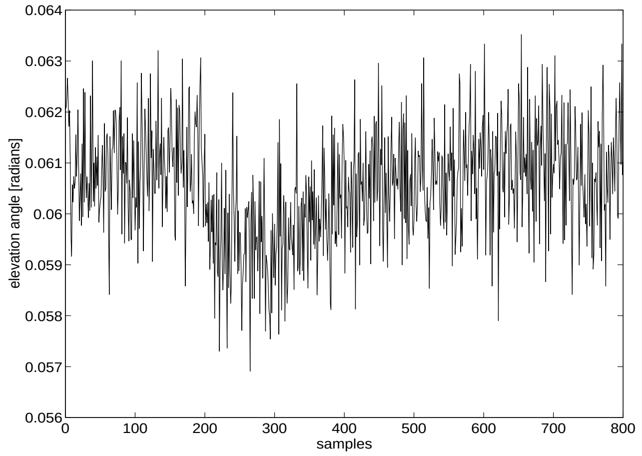


Fig. 2. Elevation Angle Measurements.

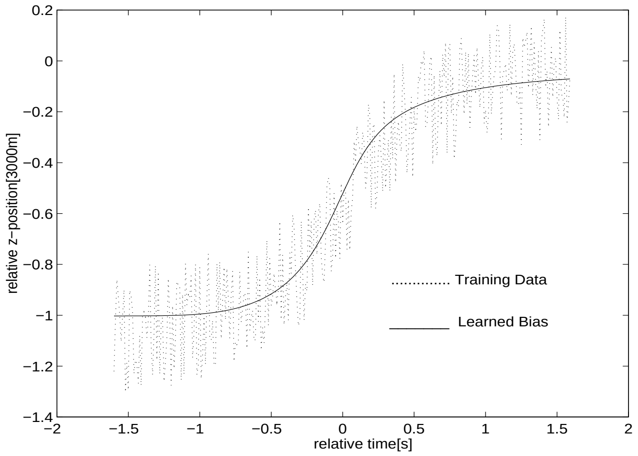


Fig. 3. Bias Function Learned From the Training Data.

to fit the empirical data of the influence of plume on the elevation angle error as a function of time. Fig. 2 shows the elevation angle measurements corrupted by Gaussian noise.

Fig. 3 shows an example of the learned bias function for a fixed velocity of the aircraft using the plume model as described above. We applied the root mean-square curve fitting algorithm to evaluate the performance of the back-propagation neural network. The normalized root mean-square error is about 0.01, indicating a small one percent error for the fit of the learned bias function to the noise-free plume model.

The impact of velocity on the accuracy of the learning process was tested by training the network for different velocities. The velocity range was subdivided into intervals, and batches of training data were generated for each interval. Fig. 4 shows the root mean-square error (RMSE) of the bias function approximation as a function of the number of velocity points. Using more than ten velocity points does not result in the significant decrease of the RMSE.

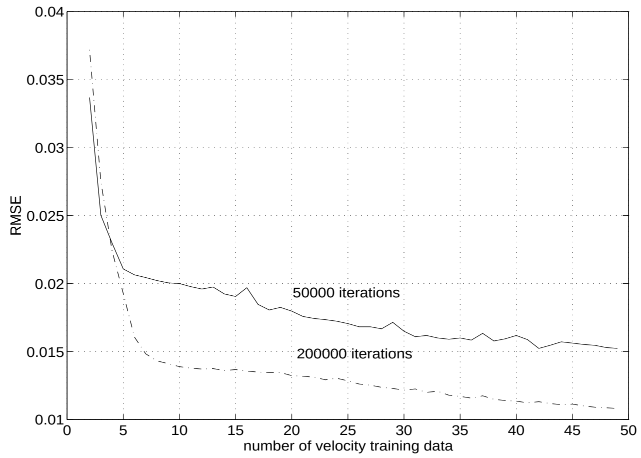


Fig. 4. The RMSE of the Learning Process.

There is no need for providing training data for a large number of aircraft velocities. This conclusion is important, because it is neither a low cost nor an easy task to collect such data. The accuracy of learning also depends on the number of iterations of the backpropagation algorithm. If more iterations are run, more exact results are obtained. However, we have to be careful not to overtrain the neural network as can easily happen if we deal with small numbers of measurements.

For the sake of simplicity, we consider aircraft landing along the centerline, that is, azimuth is centered at 0 rad.

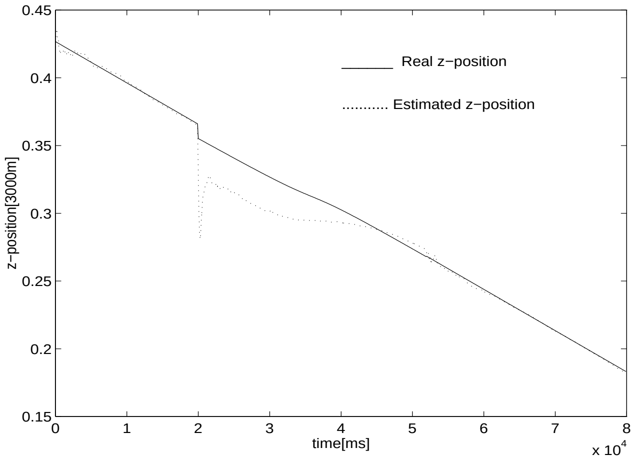


Fig. 5. Landing Aircraft Tracking Using the EKF.

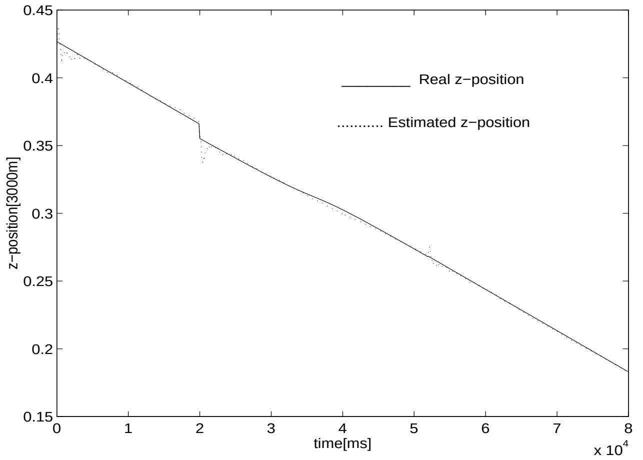
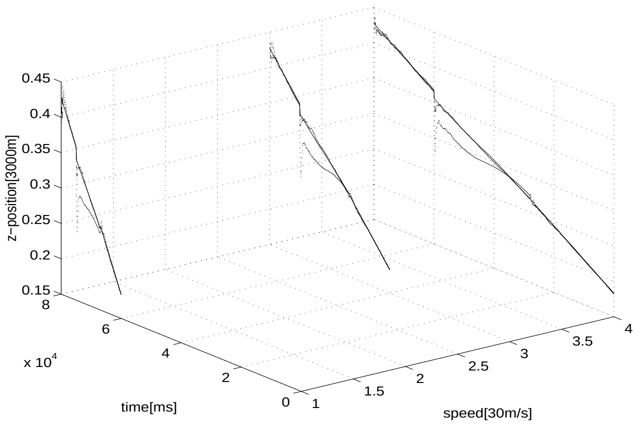


Fig. 6. Landing Aircraft Tracking Using the LTA.

The glideslope angle is constant. We assume that the plume appears after some time t_a from the moment we start to run the tracking algorithm. Additionally, we assume that for some reason (such as a gust of strong wind) the aircraft goes slightly below the glideslope at time t_a. We have only investigated the influence of plume on the z-position estimate. The influence of plume on the x-position and the y-position estimates is relatively small. Fig. 5 shows the real (the solid line) and the estimated (the dotted line) z-position of the aircraft when only the EKF is used for tracking and the FCF is not activated. The plume appears at time $t_a = 20$sec and disappears at time $t_d = 52$ sec. The EKF is tracking the plume instead of the aircraft. This results in a significant difference between the real trajectory and the estimated trajectory. Fig. 6 shows the real z-position of the aircraft and the estimated position when the LTA is used for tracking, which means that FCF is activated. A significant improvement in the tracking accuracy is achieved in this case. Fig. 7 compares the tracking results using the LTA and the EKF for different speeds of aircraft. The effect of plume on the error in the elevation angle measurements is more significant but, on the other hand, is of shorter duration in the case of faster flying aircraft.



Solid line - Real z-position from the radar

Dashdot Line - z-position estimated by the LTA

Dotted Line - z-position estimated by the EKF

Fig. 7. The Tracking Performance for Different Velocities. The LTA Track is Almost not Distinguishable from the True Track.

VI. Conclusion

A tracking algorithm based on multisensor fusion and the Kalman filter with a learning capability was proposed. This algorithm was tested on simulated data of landing aircraft. The simulation models the effects of non-gaussian noise, due to exhaust plume, on the error in elevation error measurements. Our LTA algorithm has shown to outperform the standard extended Kalman filter. The improvement in the tracking accuracy was obtained through the use of the backpropagation neural network to eliminate the effects of plume. We have also shown that, in order to achieve good results over a wide range of aircraft velocities, it is enough to train the neural network using data collected only for a few different velocities. This conclusion is economically important because, in general, it is not easy nor cost effective to collect such data. We believe that the proposed algorithm can be used in a broad range of applications including tracking of targets in cluttered environments.

References

[1] K.H. Kim, "New Filtering and Data Association Algorithms for Multiple Target Tracking," *PhD thesis,* Northeastern University, Boston, MA 02115, 1991.

[2] A. Houles and Y. Bar-Shalom, "Multisensor tracking of a maneuvering target in clutter," *IEEE Transactions on Aerospace and Electronic Systems,* AES-25, No. 2, p. 176-188, 1989.

[3] G. Biernson, *Optimal Radar Tracking Systems,* A Wiley-Interscience Publication, New York, 1990.

[4] P.L. Bogler, *Radar Principles With Applications to Tracking Systems,* A Wiley-Interscience Publication, New York, 1990.

[5] I.V. Nikiforov, *Sequential Detection of Abrupt Changes in Time Series Properties,* Ed. Naouka, 1983.

[6] I.V. Nikiforov, "Detection of abrupt changes in signals and dynamical systems," in *Lecture Notes in Control and Information Sciences,* M. Basseville and A. Benveniste, editors, New York: Springer-Verlag, 1986.

[7] L.P. Kaelbling, "Learning in embedded systems," *Technical Report TR-90-04,* Teleos Research, CA, 1990.

[8] M.M. Kokar, "Learning control: Methods, needs and architectures," in *An Introduction to Intelligent and Autonomous Control,* P.J. Antsaklis and K.M. Passino, editors, Kluwer Academic Publishers, 1993.

[9] A.M. Haimovich, J. Yosko, R.J. Greenberg, M.A. Parisi, and D. Becker, "Fusion of Sensors with Dissimilar Measurement/Tracking Accuracies," *IEEE Transactions on Aerospace and Electronic Systems,* AES-29, No. 1, p. 245-250, 1993.

[10] T. Conway, G. Gerber, M. Reiley, C. Due, and R.K. Gilbert, "Evaluation of FLIR Performance for Covert Recovery of Aircraft," presented at *Meeting of IRIS Speciality Group on Passive Sensors,* Laurel, Maryland, March 1991.

[11] F. Dufour and M. Mariton, "Tracking a 3D Maneuvering Target With Passive sensors," *IEEE Transactions on Aerospace and Electronic Systems,* AES-27, No. 4, p. 725-739, 1991.

[12] D.A. DeBell, "System Considerations for Detection and Tracking of Small Targets Using Passive Sensors," in *SPIE Proceedings on Signal and Data Processing of Small Targets Conference,* p. 180-186, Orlando, Florida, 1991.

[13] I.W. Guest and C.K. Pauw "Rank Detector Preprocessor for Glint Reduction in a Tracking Radar," *IEEE Transactions on Aerospace and Electronic Systems,* AES-29, No. 2, p. 527-531, 1993.