Multiple Target Tracking Using Support Vector Machine and Data Fusion

S.Vasuhi
Department of Electronics Engg.
MIT-Anna University
Chrompet, Chennai-600 044, INDIA
vasuhi_s@rediffmail.com

V.Vaidehi
Department of Electronics Engg.
MIT-Anna University
Chrompet, Chennai-600 044, INDIA
Vaidehi@annauniv.edu

Midhunkrishna P.R
Department of Electronics Engg.
MIT-Anna University
Chrompet, Chennai-600 044, INDIA
midhun4krishna@yahoo.co.in

Abstract — **In this paper, same target is being sensed by multiple sensors and the main objective is to classify the information into set of data produced for the same target. Once tracks are initialized and confirmed, the number of targets can be estimated; the future predicted position and target velocity can be computed for each track. Fusion is necessary to integrate the data from different sensors and to extract the relevant information of the targets. Support Vector Machines (SVMs) are generally binary classifiers and the multi class problems are solved by combining more than one SVM. This paper proposes a novel scheme for multiple target tracking using SVM classifier. The proposed scheme achieves classification by finding the optimal classification hyperplane with maximal margin. Also Kalman Filter (KF) and 1 Backscan Multiple Hypothesis Tracking (1 BMHT) are used for filtering and association respectively.**

Keywords — Multiple Targets Tracking, multi-sensor data fusion, Support Vector Machine, Multiple Hypothesis Tracking

I. Introduction

In target tracking applications, the failure of sensors reduces the received information rate and increases the error rate. Therefore, it is important to reduce the error rate and improve the system reliability by using multiple sensors and data fusion. Sensor fusion is the process of integration and extraction of desired information from two or more sensors. It is used to increase the system's accuracy, reliability and can overcome the problems of noise and sensor failure [1]. Multi-sensor data fusion refers to intelligent processing of an array of two or more sensors. Different or the same types of information from several data sources are used for classification to achieve higher accuracy and robustness. In the area of data fusion and target tracking, multi-source classification is an important problem. Classification is realized with the theory of Support Vector Machines (SVM) and it is a new generation learning system based on recent advances in statistical learning theory [3][4]. In this paper, SVM is proposed to classify two sensor data. More SVMs are needed, when more targets are identified.

In multi-target tracking, data association is a major problem; the sensors do not provide information about the origin of the measurements. Thus a key function in fusion is the association of the measurements to the targets before any estimates can be made from the measurements [2]. Data association determines the origin of each track from the received data.

Through there are several approaches available for MTT, they have problems such as, target classification, tracking target in multisensor environment and time requirement for association, when number of targets increases. This paper proposes a novel scheme for multiple target tracking and data fusion in multisensor environment, using SVM classifier, Kalman Filter (KF) and 1 BMHT

The paper is organized as follows. Section II gives introduction about SVM. Section III gives Kalman Filter equations, Section IV describes 1-Backscan multiple hypotheses tracking. Section V gives simulation results and section VI presents the conclusion.

II. Support Vector Machine

In this paper SVM is used to classify the information obtained from two independent homogeneous sensors. For classification, a line or a hyperplane is created with maximal margin between two sets of data. The hyper-plane is defined by a number of support vectors, which are a subset of the training data available for two classes. It is used to define the boundary between the two classes [3]. Each data source is treated separately and classified by SVM.

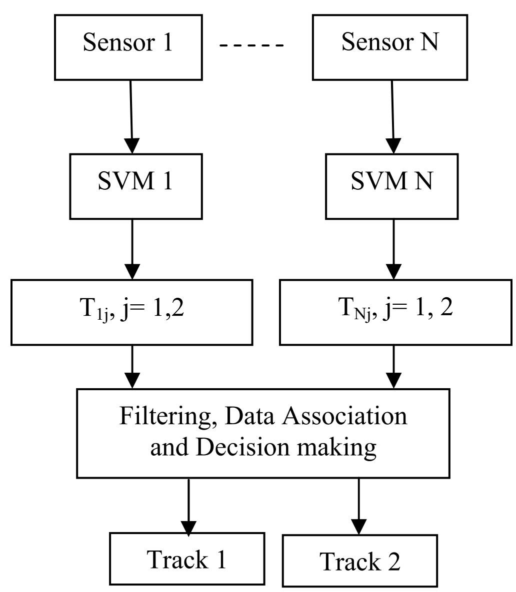

Fig.1 General block diagram of proposed work

The general block diagram of proposed work is shown in Fig.1. where T is target 1, 2, and N is the number of sensors.

In Fig. 2, a series of data points for two different classes is shown as, class A (white star) and class B (black circles). The SVM attempts to place a linear boundary between the two classes, and orients it in such a way that the margin (represented by the dotted lines) is maximized [4]. In other words, the SVM tries to orient the boundary such that the distance between the boundary and the nearest data point in each class. The boundary is then placed in the middle of this margin between the two points [5][7].

The nearest data points are used to define the margin, and are known as support vectors (represented by the gray circles and square). Once the support vectors have been selected, the rest of the feature set is not required, as the support vectors contain all the information needed to define the classifier [8].

The boundary can be expressed by

$$(w.x)+b=0, w\in R^N, b\in R \quad (1)$$

where the vector w defines the boundary, x is the input vector of dimension N and b is a scalar threshold. At the margins, where the SVMs are located, the equations for classes A and B, respectively, are

$$(w.x)+b=+1 \quad (2)$$

$$(w.x)+b=-1 \quad (3)$$

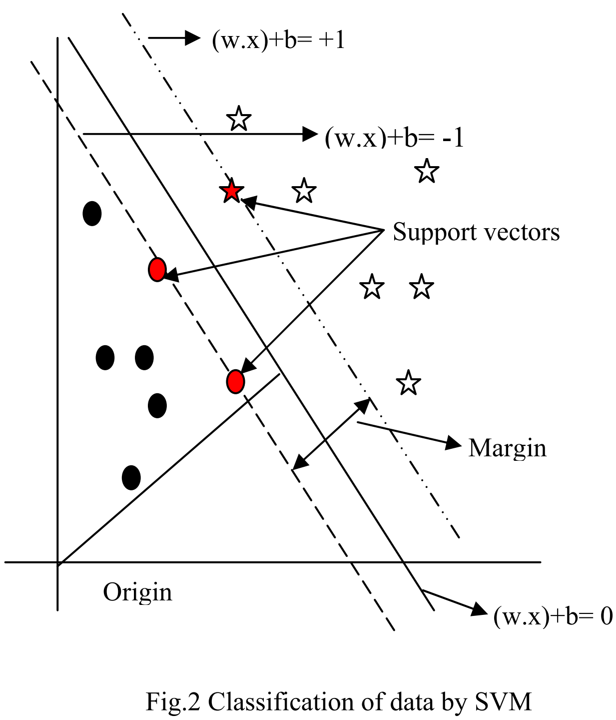

Fig.2 Classification of data by SVM

$$\sum_{i=1}^{l}\alpha_i y_i = 0, \alpha_i \geq 0 \quad (4)$$

where α_i is Lagrangian multipliers

l is the number of training sets.

When the maximal margin hyperplane is found, only points which lie closest to it have $\alpha_i > 0$ and these points are called Support Vectors (SVs). The resulting decision function is obtained as follows;

$$f(x)=sign(\sum_{i=1}^{l} y_i\alpha^*_i(x.x_i)+b^*) \quad (5)$$

The proposed fusion strategies take into account that an SVM classifier achieves classification by finding the optimal classification hyperplane with maximal margin. When margin width is low, SVM will give a general classification.

III. Kalman Filter

Suppose there are N targets and the target set is denoted by $T_N=\{1,2,....,N\}$. For the target r (r $\in$ T_N), its dynamic equation and measurement equation are given below:

$$x_t^r=F_{t-1}x_{t-1}^r+G_{t-1}v_{t-1}^r \quad (6)$$

$$z_t^r=H_t x_t^r+w_t^r \quad (7)$$

Where, x_t^r is the state vector of target r at time t and z_t^r is the measurement vector of target r at time t [6].

F_{t-1} and G_{t-1} are the system transition matrix and the input matrix at time t−1 respectively.

H_k is the measurement matrix. v_{t-1}^r is a non Gaussian driving noise for maneuvering target r and w_t^r is a zero-mean white Gaussian measurement noise vector.

IV. 1-Backscan Multiple Hypotheses Tracking

The 1-backscan Multiple Hypothesis Tracking (1 BMHT) is evolved for reducing the response time of data association such that it is lesser than that of existing correlation logics. The n-backscan has an exponential increase in response time when the number of targets increases. The Joint Probabilistic Data Association (JPDA) which forms all the possible combination of targets, shows a higher response time. The 1- backscan MHT does not require any memory to store all the previous data like n-backscan MHT. The overview of 1- backscan MHT is the same as MHT which consists of gating, filtering and prediction and hypotheses construction, evaluation and management. Gating is the first part of the correlation algorithm for eliminating unlikely observation to track pairings. A gate is formed around the predicted track position. If a single observation falls within the gate, the observation will be correlated with the track and if more than one return is within the track gate, further correlation logic is required. Filtering is used for smoothing the observations from the process and measurement noises. In prediction stage, the future position of the corrected observation is predicted to track the target [9]. Input data is collected from the signal processing unit. Hypotheses are formed as data are received and it consist of tracks whose state estimates are updated, as new data are received. The state estimates and covariance are used to form gates so that when the next data set is received, the generation of very unlikely hypotheses can be avoided. Additional hypothesis evaluation and management techniques (such as pruning, combining and clustering) are also required to limit the number of hypotheses [1].

A. Hypotheses Construction

It uses the structure branch algorithm for hypotheses tree construction. The main difference between 1-backscan MHT and n- backscan MHT lies in memory requirement. In 1- backscan MHT, present state hypothesis depends on the previous hypothesis only.

B. Bayesian Track Scoring

A relatively simple sequential technique for track scoring can be developed by applying Bayes' rule. Using Bayes' rule, the probability of correlation of true track with measurement data D is

$$p\left(T/D\right) = \frac{p\left(D/T\right) p_0(T)}{p(D)} \tag{8}$$

where, $p\left(D/T\right)$ is the probability of receiving measurement data D given that a true target is present. Also, $p_0(T)$ is the apriori probability of a true target appearing within the scan volume. The term p(D) is the probability of receiving the data D.

The Gaussian likelihood function defined as

$$g_{ij} = \frac{\exp(-d_{ij}^{2}/2)}{(2\pi)^{M/2}\sqrt{|S|}} \tag{9}$$

It is the likelihood function associated with the assignment of observation j to track i by assuming the Gaussian distribution for the residual. Similarly, $p\left(D/F\right)$ is taken to be the product of the probability of a false target return and the likelihood function ($1/V_G$) associated with the assumed uniform distribution of false returns within the volume V_G of the gated region. Thus,

$$L_k = \frac{P_D e^{-d^2/2} V_G}{P_F (2\pi)^{M/2}\sqrt{|S|}} \tag{10}$$

where d^2 is the normalized distance function and $|S|$ is determinant of the residual covariance matrix. Equation (10) can be simplified by noting that $P_F = \beta_{FT} V_G$, where β_{FT} is the false target density. Thus (10) becomes

$$L_k = \frac{P_D e^{-d^2/2}}{\beta_{FT}(2\pi)^{M/2}\sqrt{|S|}} \tag{11}$$

Taking log of equation (11) we get log likelihood score of hypotheses and given as

$$L_k = \ln\left\{\frac{P_D}{\beta_{FT}(2\pi)^{M/2}\sqrt{|S|}}\right\} - \frac{d^2}{2} \tag{12}$$

The new target probability can be defined in terms of new target density and false target density as

$$p_0(T) = \frac{\beta_{NT}}{\beta_{NT} + \beta_{FT}} \tag{13}$$

Equation (8) to (13) provide a convenient sequential scoring scheme.

C. Track and Hypothesis Evaluation

Each track has a score which is essentially the log likelihood of the hypothesis that the set of observations in the track is from the same source. The track is a collection of false alarms. The score is initially set to zero at the time of the first observation. Thereafter, upon the receipt of data on scan k, the score for track i is updated according to the relationship.

$$L_i(k) = L_i(k-1) + \Delta L(k) \tag{14}$$

where,

$\Delta L(k) = \ln(1 - P_D)$; no track update

$= \Delta L_G$; track updated

$$\Delta L_G = \ln\left\{\frac{P_D}{\beta_F (2\pi)^{M/2}\sqrt{S_t^r}}\right\} - \frac{d^{r2}}{2} \tag{15}$$

P_D = estimated probability of detection

β_F = false target density

M = measurement dimensionality.

S_t^r = residual covariance matrix

P = Kalman filter predicted value,

R = measurement error covariance matrices

H = measurement matrix

d^{r2} = normalized statistical distance function

Track scores are updated independently and the track observation histories are recorded. Then, a clustering method is used to identify interacting tracks. Within a cluster, compatible tracks are defined to be tracks that do not share observations.

D. Filtering and Prediction

Here, Kalman filter is used as the target motion is assumed to be linear and in a Gaussian environment. KF filter is a recursive estimator, so the previous time step and the current measurement are needed to compute the estimate for the current state. The Kalman filter has two distinct phases: predict and update. The predict phase uses the state estimate from the previous time step to produce an estimate of the state at the current time step. In the update phase, measurement information at the current time step is used to refine this prediction to arrive at a new, (hopefully) more accurate state estimate, again for the current time step.

Predict

The predicted state and measurement equation are given by equation (6) and (7) respectively.

Predicted estimate covariance $P_k = F_k P^r_{k-1} F_k^T + Q_k$

Update

Innovation or measurement residual $y_k = z_k - H_k x_k$

Innovation (or residual) covariance $S_k = H_k P_k H_k^T + R_k$

Optimal Kalman gain $K_k = P_k H_k^T S_k^{-1}$

Updated state estimate $x^r_k = x_k + K_k y_k$

Updated estimate covariance $P^r_k = (I - K_k H_k) P_k$

E. Track Management

Hypotheses are constructed from sets of compatible tracks. The hypothesis score is the sum of the scores of the tracks contained in the hypothesis. For given hypothesis scores, L_{Hy_j}, the probability of hypothesis $P(Hy_j)\,j$ can be computed, using all J hypotheses, using

$$P(Hy_j) = \frac{\exp(L_{Hy_j})}{[1 + \sum_{j=1}^{J} \exp(L_{Hy_j})]} \qquad (16)$$

Note that a given track can be contained in more than one hypothesis. Thus, the probability of a track is the sum of all hypotheses that contain the track. Also, similar tracks are merged.

V. Simulation Results

The constants used for calculating the score of tracks in hypothesis are

P_D = 0.9, estimated probability of detection

β_F = 0.5, false target density

M = 2, measurement dimensionality.

Fixed-Center Constant Turn-Rate Model is designed. Assume p, v and a denote the target position, velocity, and acceleration vectors, in an inertial coordinate system. A general rotational motion of a particle around a fixed center p_0 is described by the following vector equation of a rigid body motion

$$V = \Omega \times (P - P_0) \qquad (17)$$

where Ω denotes the angular velocity vector and $\times$ denotes the vector (cross) product operation. The angular velocity can be expressed in terms of the velocity and acceleration vector

$$a = \Omega \times V$$

$$\Omega = \frac{V \times a}{V^2}$$

$$\omega = \|\Omega\|$$

where ω is the turn rate

$$x^r_t = \begin{bmatrix} 1 & \text{sinwT} & \frac{(1-\text{coswT})}{w^2} \\ 0 & \text{coswT} & \frac{\text{sinwT}}{w} \\ 0 & \text{-wsinwT} & \text{coswT} \end{bmatrix} x^r_{t-1} + \begin{bmatrix} \frac{\text{wT-sinwT}}{w^3} \\ \frac{\text{1-coswT}}{w^2} \\ \frac{\text{sinwT}}{w} \end{bmatrix} v^r_{t-1} \qquad (18)$$

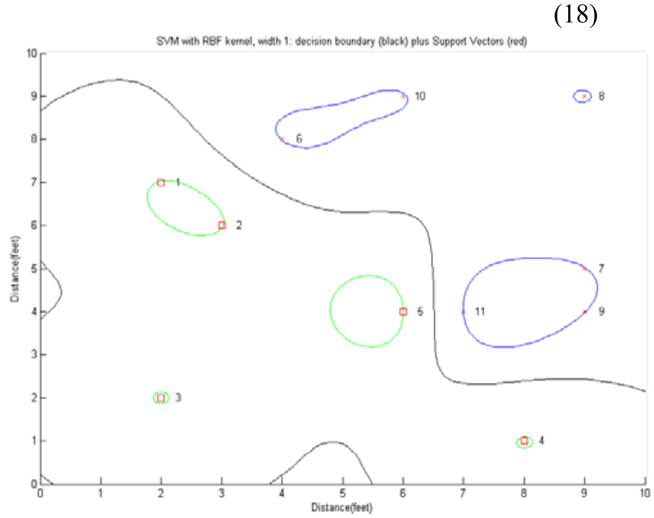

Fig.3 SVM with Radial basis function (width=1)

Fig.3 shows the tracking of target 1 by using SVM data fusion. The kernel function used in the fusion process is Radial Basis Function (RBF). With increase in width of the RBF the data fusion process will give a more accurate result.

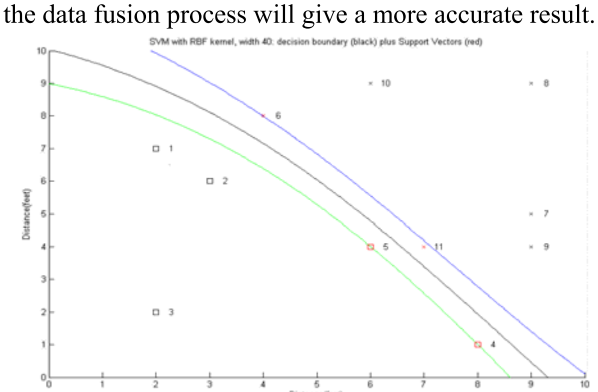

Fig..4 SVM with RBF=40 for sensor1

The width of kernel function (RBF) is taken as 40. With this high RBF, the SVM classifies the data correctly. Fig.4 and Fig.5 show the SVM classifier output for sensor 1 and sensor2 respectively.

Fig.6 shows the tracking of two crossing targets using 1 -backscan MHT with Kalman filter.

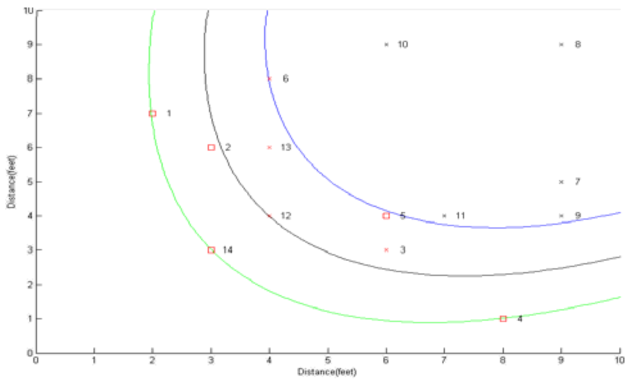

Fig.5 SVM with RBF=40 for sensor 2

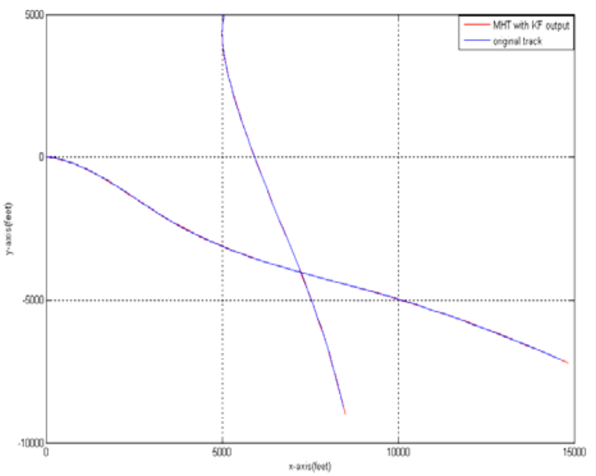

Fig.6 Simulation result of MHT with Kalman filter

TABLE I. EXECUTION TIME COMPARISON

No. of targets	JPDA (sec.)	1 Backscan MHT (sec.)	Zero Backscan MHT (sec.)
3	8.580544	19.012465	16.835849
5	31.467946	13.762308	13.014490
7	-	16.944705	17.414620
9	-	25.560077	27.549480

The execution time of the 1-backscan MHT is compared with the correlation logic Joint Probabilistic Data Association (JPDA) [10] and Zero-Backscan MHT (ZBMHT) and the results are shown in Table I. It is found that the proposed scheme gives better results in terms of execusion time with increased number of targets.

V. CONCLUSION

This paper proposes tracking of multiple targets and data fusion using SVM classifier, Kalman Filter and 1 BMHT. They are evaluated by applying them to two sensors and two targets environments. The training of binary SVM classifier is done by high RBF. When measurements are received from multiple sensors, the classified data are combined together and track the target. The combination of SVM, Kalman Filter (KF) and 1 backscan Multiple Hypothesis Tracking provides accurate tracking results. By combining more classes, larger databases from multiple sensors can be fused, but it will be more time consuming.

REFERENCES

[1]. Blackman S.S. (1986), "Multiple targets tracking with radar applications" Artech House.

[2]. Blackman S., Dempster R., Broida T. (July 1993), "Multiple Hypothesis Track Confirmation For Infrared Surveillance Systems" IEEE Transactions on Aerospace and Electronic Systems, July 1993.

[3]. Zang Xizhe,Zhao Jie,Wang Chen, Cai Hegao."Study on a SVM-based Data Fusion Method, proceedings of the, 2004 IEEE,Conference on Robotics ,Automation and Mechatronics Singapore,Dec-2004.

[4]. Gao Guohua1, Zhang Yongzhong, Zhu Yu1, Duan Guanghuang, "Data Fusion and Multi-fault Classification Based On Support Vector Machines",Joint Conference on Information Sciences (JCIS) -2006 Proceedings, Atlantis-press

[5]. Machine,Jingwen Tian and Meijuan Gao,Erhong Lu "Dynamic Collision Avoidance Path Planning for Mobile Robot Based on Multi-sensor Data Fusion by Support Vector". proceedings of the 2007 IEEE,International Conference on Mechatronics and Automation, August 5-8,2007,China.

[6]. X. Rong Li_ and Vesselin P. Jilkov_, "A Survey of Maneuvering Target Tracking: Dynamic Models" Proceedings of SPIE Conference on Signal and Data Processing of Small Targets, Orlando, FL, USA, April 2000

[7]. Hu Zhong-hui , Cai Yun-ze , LI Yuan-gui , XU Xiao-ming , "Data fusion for fault diagnosis using multi-class Support Vector Machines" Journal of Zhejiang University, SCIENCE2005

[8]. Pakorn Watanachaturaporn, Pramod K. Varshney;Manoj K. Arora, "Multisource Fusion for Land Cover Classification using Support Vector Machines", *2005* 7th International Conference on Information Fusion, pp. 614-621, 2005.

[9]. X. Rong Li and Vesselin P. Jilkov, "A survey of maneuvering target tracking Part V: Multiple model methods", IEEE Transactions on Aerospace and Electronic Systems vol. 41, no. 4, 2005.

[10] Jitendra K. Tugnait "Tracking of Multiple Maneuvering Targets in Clutter Using Multiple Sensors, IMM, and JPDA Coupled Filtering" IEEE transactions on aerospace and electronic systems, January 2004.