Robust Tracking of Maneuvering Target with Appearance Variation in Infrared Images Sequence

JIU-QING WAN, XIAO-QING ZHANG AND JIN-SONG YU
School of Automation Science and Electrical Engineering
Beijing University of Aeronautics and Astronautics
Beijing 100083, P.R. China

Tracking of maneuvering target in infrared images is a challenging problem, especially when the target under tracking experiences large appearance variations or disappears temporarily during some periods. The difficulties lie in the uncertainties in both target motion mode and target appearance, and the nonlinearity in the observation process when the images are treated as measurements directly. This paper presents a robust tracking algorithm to cope with these difficulties. We propose a mixture observation model, which can describe both the gradual intensity variation and sudden disappearance of target pixels, and use an online EM algorithm to update the model parameters. Target is tracked with the interacting multiple model particle filter (IMM-PF), where the proposed adaptive observation model is used to assign weights to the particles based on current measurement. The problems of the simultaneous update of target state and observation model and calculation of motion model likelihood are investigated. Moreover, particle number adaptation is introduced to improve the efficiency and robustness of the algorithm. Finally, we extend the algorithm to multiple targets tracking by introducing a likelihood function based on the probabilistic exclusion principle. Experimental and simulation results demonstrate the robustness of our algorithm.

Keywords: adaptive observation model, EM algorithm, interacting multiple model, particle filter, target tracking

1. INTRODUCTION

One of the main focuses in surveillance and navigation is the robust tracking of single or multiple targets in dense clutters. However, this is a rather formidable problem. In typical scenarios of practical interest, the target’s motion usually exhibits strong maneuverability which cannot be described by any specific dynamic model. This always makes the single model based tracker fail. In addition, in the situation where the underlying dynamic and/or observation process is non-Gaussian and/or nonlinear, the performance of traditional linear tracker such as Kalman filter deteriorates significantly. In fact, in imaging based tracking the observation model is a highly nonlinear mapping from the target location to a specific pixel intensities distribution in the field of view. Finally, the intensity distribution and shape of target under tracking may change randomly over time as a result of the variation of target attitude, distance and conditions of observation such as illumination and air condition, etc. Due to optical occlusion or electronic countermeasure of non-cooperative target, temporary disappearance and the reappearance of target in the field of view may be encountered during tracking process. This means that the observation
model is temporal variant and precludes the application of fixed observation model based tracker in this situation.

In this paper we focus on the problem of tracking small maneuvering target undergoing intensity variation and temporary disappearance in infrared images sequence. We propose a novel algorithm to accommodate the difficulties mentioned above, i.e., the uncertainty of the target motion model, the nonlinearity and non-Gaussianness of underlying process and the variation of observation model, in a joint manner. After a brief review of the related works in section 2, we formulate the problem in a Bayesian point of view in section 3. In section 4, we model the observation of each pixel in the target area independently by a mixture of Gaussian and uniform density and present the algorithm to update the mixing probabilities and density parameters recursively. The proposed observation model can provide enough flexibility to describe the intensity variation and even disappearances of the target under tracking. Furthermore, to improve the tracker’s performance on tracking highly maneuverable target and makes it suitable for non-Gaussian nonlinear system, in section 5 we incorporate the adaptive observation model into the interacting multiple model particle filtering (IMM-PF) framework proposed in [1], where the state probability density is approximated by the discrete sums of a set of weighed samples called particles. The adaptive observation model is updated and used to assign weights to the particles based on the current measurement, i.e. the current infrared image frame. We revise the calculation of dynamic model likelihood in [1] and introduce particles number adaptation to make the traditional IMM-PF framework more effective to our application. In section 6, to generalize our algorithm to multiple targets tracking application, we present a new likelihood function for the joint state vector of multiple targets, which precludes the possibility that a single piece of image data is used to support different targets hypotheses. Experimental and simulation results presented in section 7 demonstrate the robustness of our proposed tracking algorithm and conclusion is given in section 8.

2. PREVIOUS WORKS

For tracking maneuvering targets with nonlinear non-Gaussian model, it is necessary to combine the multiple model based method with particle filtering. Unfortunately, the discrete nature of particle representation of probability density disenable the interaction operation which is the core of the standard IMM framework. Boers [1] presents an effective way for interacting the probability density represented by weighed particles that preserves the multimodal property of the state distribution. In [1], the author applied a regular version of particle filter where the probability density function, represented by a point mass probability density on a number of grid point in the state space, is fitted to a continuous probability density function that is a sum of a prefixed number of Gaussian density function. The model conditioned particle filters are interacted by sampling the resulting Gaussian mixture model. The IMM-PF proposed in [1] is designed for the case where the measurement is a fixed linear/nonlinear function of the target state. However, in our tracking strategy, eliminating the target detection step, the image frame is directly used as measurement. Here the observation model is a highly nonlinear mapping from target location to a specific pixel distribution and cannot be described by any analytical
nonlinear function of target state. In addition, the fixed likelihood function in [1] is not suitable for tracking appearance varying target.

In imaging based tracking, the accommodation of target appearance variation is an active research topic. The natural solution to this problem is to use adaptive appearance model. There are lots of works in this direction, with different target representation and update schemes. Dawoud [4] directly used the intensity representation of target and updated the target model by calculating the weighted sum of the consecutive target image patches detected in previous frames. The values of weights were set so that the previous detected target images are forgotten gradually and the values of all weights are sum to one. The resulting composite image, called weighted composite reference function, was used to detect target in the current frame. However, giving more weight to the most recent detected target image patch seems unreasonable if the target’s appearance experienced large distortion or disappearance in the last frame, and thereby may lead to loss of track in such scenarios. Bal [3] investigated tuning basis functions for reappeared target detection problems using training based Fukunaga-Koontz transform. The tuning basis functions were used to define possible target images during target is not in the frame scene, and to detect tentative target image combined with template matching algorithm when target reappeared. The training process involved in Bal’s method, however, needs to collect training samples for both target and clutter class and to calculate the eigenstructure of the covariance matrices of the collected samples. This batch-mannered operation makes the proposed method unsuitable for real-time application. Venkataraman [6] presented a dual foreground-background target appearance model using the statistics of both foreground and background as features for target detection. Moreover, the author proposed an online feature selection scheme that can adaptively assign different weights to different features to accommodate the variation of target appearance. However, as the author pointed out, the proposed method does not show improvement when target is small, e.g., less than 50 pixels in the target area. The reason is, according to the author, that the small targets make histogram based feature extraction unreliable due to the fixed bin numbers and thus reduce the merit of feature selection.

In most of the imaging based tracking approaches nowadays, the target detection and tracking are accomplished independently in different phases. The resulting data of target detection are used as measurement in tracking phase. However, it has been pointed out in [2, 5] that this suboptimal decoupling of detection and tracking tasks may lead to poor performance in the case of low target-to-clutter ratios. In addition, in this decoupling strategy, the potential valuable target appearance information can not be used effectively in target state estimation and data association process. In [2], through the introduction of a nonlinear observation model that, when a target is present, maps a target centroid location into a spatial distribution of target pixels with a certain shape and intensity, Bruno proposed a methodology that enables direct target tracking from image sequence, eliminating any preliminary single frame detection step and integrating detection and tracking. To accommodate the target appearance variation, beside the traditional dynamic state variables such as position, velocity and acceleration, the author introduced an additional discrete-valued state variable to indicate the target’s aspect. Then a mixed state particle filter algorithm was presented for direct multiaspect target tracking from image sequences. The drawbacks of Bruno’s method are that the prior knowledge about the target template of each aspect is necessary and the continuous change in target appearance is difficult to
be modeled by the discrete aspect variable.

In this paper, we proposed a robust target tracking method based on IMM-PF with adaptive observation model. Our algorithm has the same basic IMM-PF structure as in [1] but differentiates their works in the following aspects: (1) In our IMM-PF framework, the image data are used directly as measurement and the potential valuable target appearance information can be used in a systematical way. The observation process here is a highly nonlinear mapping from target state to pixels intensity. We also solve the problem of dynamic model likelihood evaluation brought by this new observation process. (2) We use a more informative observation model to describe the appearance variation. Specifically, we proposed a two component mixture observation model which can describe both the gradual intensity variation and sudden disappearance/reappearance of target pixels. Our observation model is more flexible than that used in [2] where appearance variation is handled by a discrete valued random variable and gets rid of the need for the prior knowledge of target aspects required in [2]. (3) The online EM algorithm proposed in [7] is used to update the observation model parameter recursively. To incorporate this update algorithm into the IMM-PF framework and to enable simultaneous update of target state and observation model, we approximate the current state required in the EM algorithm by the predicted state which maximizes the likelihood conditioned on the previous model parameters. Moreover, some modification has been made on the original EM algorithm in [7] to improve its robustness. (4) Particle number adaptation is introduced in our IMM-PF framework. For each model conditioned particle filter, the number of particles to be generated in the resampling process is adjusted automatically according to the covariance of the state prior density. Compared with [1] where the particle number is fixed, our algorithm is more efficient and robust. (5) We generate our algorithm to multiple targets tracking by introducing the joint state vector and replacing the likelihood function for single target state by a exclusion principle based likelihood function for the joint state which can preserve the joint tracker coalesce onto one of the targets under tracking after multiple targets crossed. The new likelihood function is also parameterized and can be updated by the online EM algorithm based on measurement data.

3. PROBLEM FORMULATION IN A BAYESIAN FRAMEWORK

In this section, we formulate the problem of tracking maneuvering target undergoing appearance variation. Let $\Phi_k$ denote the target state at time instant $k$, consisting of variables describing target motion such as position, velocity, acceleration, and angular rate, etc.. The evolution of target state can be modeled by a stochastic discrete time state space transition equation

$$
\Phi_k = f(\Phi_{k-1}, w_{k-1}, S_k),
$$

where $w_k$ is a i.i.d. system noise sequence and $f$ is a possibly nonlinear mapping from space $\mathbb{R}^{n_\Phi} \times \mathbb{R}^{n_w}$ to space $\mathbb{R}^{n_\Phi}$, $n_\Phi$ and $n_w$ are dimension of target state and system noise respectively. The motion of maneuvering target cannot be specified by any single fixed model, therefore it is assumed that a set of candidate models are available at any time instant and the discrete valued random variable $S_k$ is introduced to indicate the model.
ROBUST TRACKING OF TARGET WITH APPEARANCE VARIATION

takes effect at time $k$. The domain of possible value of $S_k$ is $\{1, \ldots, M\}$, where $M$ is the number of dynamic models under consideration. Moreover, $S_k$ is assumed to follow a first order Markov chain with transition matrix $\delta$ and initial distribution $\mu_0$. Given the statistics of system noise, the above nonlinear non-Gaussian state space model (1) specifies the predictive transition density $p(\Phi_k | \Phi_{k-1}, S_k)$.

At time instant $k$, a new measurement $Z_k$ is available, which is related with the target state through the following nonlinear function

$$Z_k = h(\Phi_k, v_k, \Theta_k),$$  \hspace{1cm} \text{(2)}

where $v_k$ is a i.i.d. observation noise sequence and $h$ is a highly nonlinear mapping from target state space $R^{n\Phi} \times R^{n\nu}$ to measurement space $R^{nZ}$, $n_Z$ and $n_{\nu}$ are dimension of measurement and measurement noise respectively. In our application, the measurement is the image frames themselves. To model the target appearance variation, the random variable $\Theta_k$ is employed to parameterize the mapping process. Given the statistics of measurement noise, the above observation model (2) specifies the likelihood function of the current measurement conditioned on the current state $p(Z_k | \Phi_k, \Theta_k)$. We define the parameterized likelihood function and study the parameter estimation in section 4. The conditional dependency of the involved random variables is graphically depicted in Fig. 1.

![Dependency of involved random variables. Squares depict discrete variables, circles depict continuous variables, shaded circles depict observable variables.](image_url)

From the Bayesian point of view, the posterior density $p(\Phi_k | Z_1:k)$ constitutes the complete solution of the tracking problem, from which the interested statistics such as mean and covariance can be calculated. The object of tracking algorithm is to sequentially calculate $p(\Phi_k | Z_1:k)$, based on the previous posterior density $p(\Phi_{k-1} | Z_1:k-1)$ and current available measurement $Z_k$. Usually, previous posterior density is multiplied by the predictive transition density firstly and the previous state $\Phi_{k-1}$ is integrated out to generate the prior density $p(\Phi_k | Z_{1:k-1})$, which is then corrected to give $p(\Phi_k | Z_{1:k})$ based on current measurement via Bayes’ rule. The difficulty lies in the existence of the nonlinear mapping and non-Gaussian noise in Eqs. (1) and (2), and the introduction of two nuisance random variables $S_k$ and $\Theta_k$. In section 5, we approximate the belief state by a set of weighted particles, and detail the IMM-PF framework for recursive estimation of the posterior density, accommodating the above mentioned difficulties.
4. ADAPTIVE OBSERVATION MODEL

In target tracking based on infrared imaging, at each sampling instant the observation data is 2-D digital image of size $N_h \times N_v$, which is referred to as a frame. We represent the intensity values of the $k$th frame as a $N_h \times N_v$ matrix $y(k)$. The observation process is, when the target is present, a highly nonlinear mapping from the target centroid coordinates contained in the target state vector $\Phi(k)$, to a special distribution of image pixels with certain intensities. It should be noticed that the state variables are continuous-valued and must be discretized by the image resolutions before mapping into the digital image; we refer the reader to [2] for detailed description for the discretization process. We might include in the state vector the size or the attitude of target with respect to a known reference, such as in [6-8]. But we concentrate on small target tracking here and handle the attitude change by appearance model adaptation.

4.1 Target Model

To write down a statistic model of the observation $y(k)$ expending on the target state $\Phi(k)$, we first introduce a window function $w_l(k)$ that defines the spatial support of target, i.e., the pixels in a rectangular area of certain size centered around the location of target. The size of the window is chosen so that it can cover the maximum possible target of interest. The window is sliding over the observed image $y(k)$ according to the variable $l$ indicating target location. The resulting sub-image after applying $w_l(k)$ to $y(k)$ is denoted as $z(k) = w_l(k)y(k)$. Before we write the likelihood function, the variables involved in observation model should be translated to vector form. The vector equivalent representation of matrix $y(k)$ and $z(k)$ is obtained by scanning the matrix row by row and sequentially stacking the scanned rows into a long vector. Assuming that the pixels of target and background are statistically independent and we have no further knowledge about the distribution of background, the likelihood function can be written as

$$p(y(k) | \Phi(k)) = p(w_l(k)y(k) | \Phi(k)) \times p((1 - w_l(k))y(k) | \Phi(k)) \propto p(w_l(k)y(k) | \Phi(k))$$

where the background pixel distribution follows a uniform pdf and hence the term $p((1 - w_l(k))y(k) | \Phi(k))$ is a constant.

Next, we will parameterize the likelihood function $p(z(k) | \Phi(k))$. Usually, due to variations in target attitude, distance and conditions of observation, the target intensity is expected to be changing smoothly. However, suddenly shape changes or entirely disappearance of target is also expected caused by the events such as optical occlusion or electronic countermeasure. It should be noticed here that the shape variation of target can be considered as a partially disappearance or reappearance of pixels in the target region at certain time instances. We further assume that the $d$ pixels in the window’s support region are statistically independent of each other, and denote the intensity of the $j$th pixel in the region as $z_j(k)$. The phenomena in target appearance variation motivate us to model the target with a two component mixture density. In fact, the mixture observation model
has been used in others references. Zhou [9] used the image intensity directly as measurement data and modeled the target appearance with a three component Gaussian mixture. However, this model cannot explain target disappearance and occlusion is handled by robust statistics in his work. Jepson [7] proposed a generative appearance model which was formulated as a mixture of three components, namely, a stable component, a 2-frame transient component and an outlier process. But the measurement data used in [7] is the complex valued responses of steerable pyramid filters. The first component in our mixture model is a Gaussian density $N(z_j(k); m_j(k), \sigma_j^2(k))$ capturing the gradually changes in target intensity, where $m_j(k)$ and $\sigma_j^2(k)$ are slowly varying functions specifying the mean and variance of Gaussian density. When pixel in target region is absent, its intensity is determined by background. Thus it can be modeled by the second component density $U(z_j(k))$ which is a uniform distribution over the observation domain. The above two components are combined into a probabilistic mixture model

$$p(z(k) \mid \Phi(k), \Theta(k)) = \sum_{i \in \{p, a\}} \pi_{j,i} p_i(z_j(k) \mid \Phi(k), \theta_j(k))$$

$$= \pi_{p,j}(k) N(z_j(k); m_j(k), \sigma_j^2(k)) + \pi_{a,j}(k) U(z_j(k))$$

where $\pi_{j,i}, i \in \{p, a\}$ are the probabilities that the target pixel is present or absent, and $\theta_j(k)$ denote the mean and variance of the Gaussian component in the model of the $j$th pixel at time $k$. From the independency assumption of pixels, the likelihood function of whole target region can be written as

$$p(y(k) \mid \Phi(k), \Theta(k)) = \prod_{j=1}^d \left( \sum_{i \in \{p, a\}} \pi_{j,i} p_i(z_j(k) \mid \Phi(k), \theta_j(k)) \right)$$

where $\Theta(k) = \{\pi_{p,j}(k), \pi_{a,j}(k); m_j(k), \sigma_j^2(k)\}_{j=1}^d$.

### 4.2 Model Parameter Adaptation

To accommodate the target appearance variation, the likelihood function should be updated online, that is, the parameters $\Theta(k)$ in likelihood model should be estimated recursively based on all available observations up to current time instant. We use the online EM algorithm proposed in [7] to estimate these parameters. Based on the assumption of independency of pixels in target region, we can estimate the parameters of the likelihood of each pixel $\Theta_j(k)$ independently. For purpose of simplicity, we drop the subscript $j$ for few lines.

Suppose the past data are exponentially forgotten with respect to their contribution to current observation model. In other words, the data are considered under an exponential envelope located at the current time, i.e., $S_i(t) = \alpha e^{-t/\tau}$, for $t \leq k$, where the time constant $\tau$ controls the attenuation speed of the envelope, and $\alpha = 1 - e^{-1/\tau}$ so the envelope weights $S_i(t)$ sum to 1. The EM algorithm [14] is used to find the ML estimate of the parameters in an iterative way based upon the history available data enveloped by $S_i(t)$, that is
\[ \hat{\Theta}(k) = \arg \max_{\Theta} \sum_{i=k}^{\infty} S_i(t) \log(p(y(t) \mid \Phi(t), \Theta(k))). \] (6)

When the envelope slides across time, we obtain parameters estimate at different time instances. For the purpose of self-contained of this paper, we sketch the estimation algorithm in [7] briefly as follows:

Firstly, given a current guess of the parameters \( \hat{\Theta}(k) \), we calculate the ownership probabilities of the observation \( y(t) \)

\[ o_{i,k}(t) = \frac{\pi_{i,k}^p(k)p_i(y(t) \mid \Phi(t), \hat{\Theta}^p(k))}{\sum_{i=\{p,a\}} \pi_{i,k}^p(k)p_i(y(t) \mid \Phi(t), \hat{\Theta}^p(k))} \] (7)

for \( i \in \{p, a\} \).

Then some intermedia quantities are calculated

\[ \hat{\mathbf{q}}_i(k) \approx \sum_{i=k}^{\infty} S_i(t) o_{i,k}(t) \]
\[ = ao_{i,k}(k) + (1-a)\hat{\mathbf{q}}_i(k-1), i \in \{p, a\}, \]
\[ \hat{\mathbf{M}}_{(j)}(k) \approx \sum_{i=k}^{\infty} S_i(t) z^j(t) o_{p,i}(t) \]
\[ = az^j(k)\hat{\mathbf{M}}_{(j)}(k-1), j \in \{1, 2\}. \] (8)

And the parameters estimates are given by

\[ \hat{n}_i(k) = \frac{\hat{\mathbf{q}}_i(k)}{\sum_j \hat{\mathbf{q}}_j(k)}, i \in \{p, a\}, \]
\[ \hat{m}(k) = \frac{\hat{\mathbf{M}}_{(1)}(k)}{\hat{\mathbf{q}}_p(k)}, \sigma^2 = \frac{\hat{\mathbf{M}}_{(2)}(k)}{\hat{\mathbf{q}}_p(k)} - \hat{m}^2(k). \] (9)

In Eq. (8), the factor \( \alpha \) controls the sensitivity of the model to new coming data and takes value between 0 and 1. Larger \( \alpha \) implies the model has a rapid response to the new observation whereas smaller \( \alpha \) make the model tend to insist on the previous values of parameters.

However, in practice we find that the proposed model is too flexible that it easily overfits to some background patterns and results in loss of tracking. Thus we exert some constraints on the model and the updating process to avoid these overfitting phenomena. Firstly, with both varying mean and variance of Gaussian model, when a new intensity data is available, the original update algorithm tends to modify the value of variance rather than that of mean. Thereby it is always the case that after few iterations, the value of variance diverges. We modify the original algorithm in that, instead of updating both
the mean and variance of intensity as in [7], we fix the value of variance and update the mean only. Another problem in the original update algorithm is that, as the pixels in target region are dealt with independently, when target is absent, the model is easily fit to some background region that are partially similar to target, resulting in very high likelihood of the particles fall in this region. To handle this problem, in our implementation, if a predefined number of pixels’ mixing probabilities $\pi$ are less than a specific threshold, then the mean parameters corresponding to all pixels stop updating, that is, $\pi$, $i \in \{p, a\}$ and $M(1)$ are updated as usual, but the values of $m$ are kept unchanged. It has been shown in simulation that these modifications lead to very good performance in observation model update.

4.3 Model Initialization

To initialize the observation model, for each pixel in target region, we set $m_j(0) = r_j$, where $r_j$ is the $j$th element of the vectorized initial template of target; set $M_j(0) = m_j(0)$, $q_{p,j}(0) = 0.9$, $q_{a,j}(0) = 0.1$ for all pixel elements; set the variance as constant $\sigma_j^2 = 10$.

5. IMM-PF BASED TRACKING

In this section, we investigate the incorporation of the adaptive observation model into the interacting multiple model particle filtering framework proposed in [1]. In the IMM-PF proposed in [1], interactive multiple dynamic models were employed to efficiently propagate particles representing target state for highly maneuverable target tracking. We use the approach in [1] to implement the interaction between different model based particle filters, where the Gaussian mixture models fitted by particles of each filter are combined with mixing probabilities calculated as in classical IMM and at the succeeding iteration each filter is reinitialized by samples drawn from the resulting GMM combination. However, three problems arise when we use the original IMM-PF in [1] with adaptive observation model treating image as measurement directly. Firstly, it must be decided when and how to update the observation model in the IMM-PF framework. It can be seen in Eq. (7) that the evaluation of ownerships and thus the update of observation model needs the knowledge of current target state. But the current target state is unknown and need to be estimated based on the new observation. Traditionally an approximation of current state, e.g., the previous state estimate [8-10], is usually used for observation model update. However, for small maneuvering target tracking, the target state may change significantly from frame to frame. We have shown in simulations that the approximation used in traditional method always leads to diverge in observation model update. In this paper, we propose another state approximation strategy based on the predicted state instead of the previous state estimate, which will be detailed later. Secondly, the calculation of dynamic model likelihood is a crucial issue for multiple model interaction. In [1], as the observation model can be written in analytical form and the dimension of measurement is rather low, the authors calculate the model likelihood by evaluation of innovation probability based on the Gaussian assumption of the innovation density. In their approach the mean and covariance of the Gaussian model were estimated based on the predicted observations set. However, in our application the measure-
ment is the image and thus has very high dimension. It is very difficult to obtain an estimate of mean and covariance of the innovation density that is accurate enough for model likelihood evaluation. In this paper, we derive another way to evaluate the model likelihood by numerical integral approximation. Thirdly, in [1] the number of particles is fixed for each model based particle filter in every iteration. However, it seems more reasonable that if the covariance of particle prior is large, more particles are needed to cover the validation region, while conversely fewer particles are preferred to save computational resource if the covariance is small. Therefore, in our algorithm, we introduce the adaptation of particle number to improve its robustness and efficiency. The IMM-PF can be divided into four stages, which we will discuss sequentially concerning about the problems mentioned above.

Let $M$ denote the number of dynamic models considered, and $\mu(j)$ is the probability that model $j$ takes effect at time $k$, i.e.,

$$\mu(j) = \Pr(S(k) = j), j = 1, \ldots, M.$$ (10)

Suppose the model are switching with the prior probability

$$\delta_{ij} = \Pr\{S(k) = j \mid S(k - 1) = i\}, i, j = 1, \ldots, M.$$ (11)

The initial particles for each model based filter is drawn from the Gaussian density $p_0(\Phi(0))$, the mean and covariance of which are determined based on prior knowledge of target state. The steps of IMM-PF in the $k$th iteration are as follows:

Step 1: Model interaction
The model conditioned state pdfs from previous iteration are mixed to obtain the prior state pdf for current iteration. The mixing probabilities are calculated as

$$\mu_{ij}(k - 1) = \Pr\{S(k - 1) = i \mid S(k) = j, Y^{k-1}\}$$

$$= \frac{\delta_{ij} \mu_i(k - 1)}{\sum_l \delta_{il} \mu_l(k - 1)}, i, j = 1, \ldots, M$$ (12)

where $Y^{k-1}$ is the observation history up to time $k - 1$. The prior state pdf corresponding to the $j$th filter is

$$p_0^j(\Phi_0^j(k - 1) \mid Y^{k-1}) = \sum_i \mu_{ij}(k - 1)p^i(\Phi^i(k - 1) \mid Y^{k-1}), j = 1, \ldots, M$$ (13)

where $p^i(\Phi^i(k - 1) \mid Y^{k-1})$ is the previous posterior state pdf from the $i$th model based filter.

Step 2: Model conditioned filtering
One particle filter is used for each model to update the model conditioned state density. At the same time, the observation model is updated based on the new available measurement data. In addition, the likelihood of each model is evaluated. For $j = 1, \ldots, M$: 
• Determine the number of particles to be sampled $N(k)$. The covariance of the state prior density $p_{i}(\Phi(k-1) | Y^{k-1})$ is

$$
\hat{P}_i^{j}(k-1|k-1) = \sum_{i} \mu_{ij}(k-1)\{\hat{P}^i(k-1|k-1) + (\hat{\Phi}^i(k-1|k-1) - \hat{\Phi}_0^{ij}(k-1|k-1))^2\},
$$

where $\hat{\Phi}^i$ and $\hat{P}^i$ is the state and covariance estimate from the $i$th filter in the last iteration and can be calculated by Eqs. (23) and (24), and $\hat{\Phi}_0^{ij}$ is given by

$$
\hat{\Phi}_0^{ij}(k-1|k-1) = \sum_{i} \mu_{ij}(k-1)\hat{\Phi}^i(k-1|k-1).
$$

We adjust the particles number as follows

$$
N(k) = \max(\min(\varepsilon \hat{P}_i^{j}(k-1|k-1), N_{\max}), N_{\min}),
$$

where $\varepsilon$ is a scale constant, $N_{\min}$ is the lower bound to maintain a reasonable particle number and $N_{\max}$ is the upper bound to constrain the computational load. For clarity we denote $N(k)$ by $N$ in the following part of this section.

• Draw $N$ samples from state prior probability density $p_{i}^{j}(\Phi(k-1) | Y^{k-1})$ as the initial particles set for particle filter in the current iteration

$$
\{\Phi^{j}(k-1|k-1)\}_{l=1,\ldots,N}.
$$

Later we can see that the density $p_{i}^{j}(\Phi(k-1) | Y^{k-1})$ is a Gaussian mixture model.

• For each dynamic model considered, we specify the predictive transition density $p(\Phi_k | \Phi_{k-1}, S_k = j)$, then draw $N$ predictions of the current samples from this distribution

$$
\{\Phi^{j}(k|k-1)\}_{l=1,\ldots,N}.
$$

• Update the parameters of the mixture observation model based on the current available measurement by the online EM algorithm discussed in section 4.2, i.e.,

$$
\Theta^{j}(k) = update(\Theta^{j}(k-1), \Phi^{j}(k), y(k)).
$$

But $\Phi^{j}(k)$ is unknown at the moment and we make the approximation as follows: from the state predictions Eq. (18), we choose the one that maximizes likelihood function conditioned on the previous model parameters as the current state approximation. That is,

$$
\Phi^{j}(k) \approx \arg \max_{j} p(y(k) | \Phi^{j}(k|k-1), \Theta^{j}(k-1)).
$$

• Evaluate the likelihood of current observation with respect to each predicted samples by the updated likelihood function
\[
\tilde{w}_j^l(k) = p(y(k) \mid \Phi_j^l(k \mid k-1), \Theta_j^l(k)).
\] (21)

After normalization we get the weight of each samples \(w_j^l(k), l = 1, \ldots, N\). Now the weighted samples set is a discrete approximation of the true posterior density of the state conditioned on model \(j\).

- The discrete approximation of the posterior density cannot directly be merged as in the standard IMM algorithm. Following [1], we fit the posterior density to a continuous probability density function that is a mixture of Gaussian density function

\[
p_j^l(\Phi_j^l(k) \mid Y^k) = \sum_l w_j^l(k) \mathcal{N}(\Phi_j^l(k); \Phi_j^l(k \mid k-1), \nu^j \tilde{P}_j^l(k \mid k))
\] (22)

where \(\nu^j = 0.5N^{2d_j}\), and \(d_j\) is the dimension of the state space. The mean and covariance matrix can be calculated from the samples set as follows

\[
\Phi_j^l(k \mid k) = \sum_l w_j^l(k) \Phi_j^l(k \mid k-1),
\] (23)

\[
\tilde{P}_j^l(k \mid k) = \sum_l w_j^l(k)(\Phi_j^l(k \mid k-1) - \Phi_j^l(k \mid k))(\Phi_j^l(k \mid k-1) - \Phi_j^l(k \mid k))^T.
\] (24)

Note that the posterior density becomes a continuous function now and the merging operation upon these distributions in the interaction and combination stage can be implemented by standard sampling approach on Gaussian mixture.

- Model likelihood calculation. Here, we have the likelihood function of the current observation given the current state. By the assumption of conditional independence of current observation \(y(k)\) with previous observations \(Y^{k-1}\) and dynamic model \(S(k)\) given \(\Phi(k)\), we obtain

\[
p(y(k) \mid \Phi(k), S(k), Y^{k-1}) = p(y(k) \mid \Phi(k), S(k), Y^{k-1}).
\] (25)

It can be shown that the likelihood of each model is

\[
\Lambda_j(k) \triangleq p(y(k) \mid S(k) = j, Y^{k-1}) = \int_{\Phi(k)} p(y(k) \mid \Phi(k), S(k) = j, Y^{k-1}) p(\Phi(k) \mid S(k) = j, Y^{k-1}) d\Phi(k)
\] (26)

where \(p(\Phi(k) \mid S(k) = j, Y^{k-1})\) is just the predicted state distribution of the \(j\)th model, which is represented by the samples set Eq. (18). Therefore, the likelihood of each model can be approximated by
\[ \Lambda_j(k) = \int_{\Phi(k)} p(y(k) \mid \Phi(k)) \sum_{i} \frac{1}{N} \delta(\Phi(k) - \Phi_i(k)) \, d\Phi(k) \]
\[ = \frac{1}{N} \sum_{i} p(y(k) \mid \Phi_i(k)). \]  

**Step 3:** Model probability update
The model probabilities can be updated based on the likelihood of each model, that is

\[ \mu_j(k) = \frac{\Lambda_j(k) \sum_{i} \delta_j \mu_i(k-1)}{\sum_{i} \Lambda_i(k) \delta_j \mu_i(k-1)}, \quad j = 1, \ldots, M. \]  

**Step 4:** Combination
By combination of the pdf from all the model based filters, the posterior probability density of the state can be obtained

\[ p(\Phi(k) \mid Y^k) = \sum_{j} \mu_j(k) p(\Phi_j(k) \mid Y^k). \]  

And the estimates of target state and covariance are

\[ \hat{\Phi}(k \mid k) = \sum_{j} \mu_j(k) \hat{\Phi}_j(k \mid k), \]  
\[ \hat{P}(k \mid k) = \sum_{j} \mu_j(k) \{ \hat{\Phi}_j(k \mid k) + (\hat{\Phi}_j(k \mid k) - \hat{\Phi}(k \mid k))(\hat{\Phi}_j(k \mid k) - \hat{\Phi}(k \mid k))^T \}. \]

### 6. Extension to Multiple Targets Tracking

In this section, we extend the proposed algorithm to multiple targets tracking application. The first problem need to be addressed is how to choose the state space for multiple targets tracking [17]. We could use a unique particle filter with single targets state space and represent the multiple targets belief state as a multimodal distribution. However, this strategy seems inappropriate as the particle tracking an occluded target get very small weights and are therefore easily discarded during the resampling step. This problem can be handled by using one particle filter per target and tracking multiple targets independently. However, it is always the case that, especially when the targets being tracked are with similar or even identical appearance, trackers of different targets coalesce onto the same best-fitting target after their trajectories crossed each other. This is due to the mistakes in data association and thereby the misinterpretation of the measurements. Many methodologies have been developed to deal with this problem, and we will address this point shortly later. Another choice of state space, as we use in our algorithm, is to use particles whose dimension is the sum of those of the individual state space corresponding to each target. Each of these concatenated vectors then gives jointly a representation of all targets. Formally, the targets state is \( \Phi(k) = (\Phi_1(k); \Phi_2(k)) \). Here we con-
sider two targets only for clarity. The evolution of joint targets state is governed by independent dynamic model, that is

\[
\begin{pmatrix}
\Phi_1(k) \\
\Phi_2(k)
\end{pmatrix} = \begin{pmatrix}
f_1(\Phi_1(k-1), w_1(k), S_1(k)) \\
f_2(\Phi_2(k-1), w_2(k), S_2(k))
\end{pmatrix}.
\]

(32)

The most well known solution for data association problem in multiple target tracking is the joint Probabilistic Data Association (PDA) filter [12, 13]. The basic characteristic of joint PDA filter is the joint calculation of probabilities of data association events with the requirement that at any time instant, no two tracks are associated with the same single measurement. In imaging based multiple targets tracking, this requirement is widely employed. John proposed a probabilistic exclusion principle for tracking multiple objects in [18], where a new observation model is given which does not allow two targets to merge when their dynamic states become similar, but instead, continues to interpret the data in terms of two targets. Our method use the basic idea of [18], but the specific form of observation model is quite different with it. Specifically, we evaluate the likelihood function of the joint state vector as

\[
p(z(k) | \Phi_1(k), \Phi_2(k)) = \begin{cases} 
U & |\Phi_1(k) - \Phi_2(k)| < \varepsilon \\
p(z(k) | \Phi_1(k)) p(z(k) | \Phi_2(k)), & \text{else}
\end{cases}
\]

(33)

With this likelihood, the single piece of image data cannot be used to reinforce the mutually exclusive hypotheses simultaneously and hence preclude the possibility that two tracks coalesce onto the same single target. We omit the observation model parameters in Eq. (33) for clarity. In fact, the multiple targets observation model can be adapted to targets appearance by substitute Eq. (3) with Eq. (33) in the model update process.

7. SIMULATION RESULTS

7.1 Simulation Data Generation

Generation of the simulation data for algorithm evaluation consists of three subtasks, i.e. creating background image, defining target moving trajectory, defining target appearance evolution. To make our simulation as realistic as possible, we use infrared image captured by real infrared camera as the background, and assume the target moves within the background images with specific trajectory and intensity profile. The size of background image is 700 × 360 pixels. The trajectories of target motion are generated as follows: a number of fixed points are set by user in the image plane firstly and continuous target trajectory is generated by fitting these predefined points with spline functions. It should be notice that the resulting target trajectory cannot be described by any specific dynamic model but, as illustrated in our simulations, the target can be tracked successfully by the combination of multiple model based filters. To simulate the variation of target appearance, we introduce another set of signals (we call them driving signals in this paper)
specifying the intensities of each pixel in the target area at each time instant. The values of the driving signals are taken in the domain from 0 to 255. The driving signal is the sum of a user defined slowly varying signal indicating the evolution of target appearance and a colored Gaussian noise explaining the uncertainty in measurement process. During certain periods of time the values of the driving signal are set to \(-1\), implying that the corresponding pixels are disappeared, and these pixels are replaced by the background pixels at the same location. Setting all the driving signals to \(-1\) during some period means the target disappeared entirely, otherwise setting part of the driving signals to \(-1\) simulates the partial disappearance or shape variation of targets. An example of driving signal is shown in Fig. 2.

![Driving Signal Example](image.png)

Fig. 2. Example of driving signals. Each subplot defines the evolution of one pixel intensity. When the value of driving signal takes \(-1\), the pixel intensity is determined by background.

To generate the synthetic image frames to test our algorithm, at each time instant, we find the target location in the background image according to target trajectory data, and replace the background pixels in the target area by pixels with intensities determined by the driving signals. It should be noticed that through this approach, we can generate infrared image sequences which are very close to the image data captured by real infrared camera. Furthermore, the flexibility of this approach provides great convenience in algorithm evaluation.

### 7.2 Dynamic Models

We use CA (constant acceleration) and Singer model [15] in the IMM framework.
CA model assumes that the acceleration is a process with independent increments, while Singer model assumes that the acceleration is driven by a zero mean first order stationary Markov process. For a choice of the maneuver time constant value between 0 and infinite, the Singer model describes an intermediate motion between of constant velocity and constant acceleration. The states vector for both models consists of the position, velocity and acceleration in the two dimensions of image plane, \( i.e., (x, \dot{x}, y, \dot{y})^T \). We further assume the mutual independency of state variables in different dimensions, thus we can write dynamic model independently for each image dimension. For detailed descriptions of these dynamic models, we refer the reader to [15].

The initial probabilities of the two models involved are set to \( \mu_0 = (0.5, 0.5) \), and the mode transition matrix is given by

\[
\delta = \begin{pmatrix}
0.9 & 0.1 \\
0.1 & 0.9
\end{pmatrix}
\]

In simulation, the two model based particle filters are both initialized with particles sampled from the Gaussian distribution centered on the initial target state determined by the trajectory data in the time instants 0, 1, and 2. The initial particle number is set to 200. The target region consists of \( 3 \times 3 \) pixels (the algorithm can be applied on larger target region; however, we choose \( 3 \times 3 \) for clarity).

7.3 Single Target Tracking

We use both simulated and real infrared image sequence data to evaluate the performance of different single target tracking algorithm. We demonstrate the performance of our adaptive observation model based tracker in single target tracking by comparing it with: (1) algorithm using fixed observation model based on the same IMM-PF framework as ours; (2) classical interacting multiple model probabilistic data association (IMM-PDA) tracker.

The tracking results are shown in Figs. 3 and 5. Fig. 3 shows the results on simulated data where the driving signals for each pixel in the target region are defined by Fig. 2. From Fig. 2 we can see that the values of the pixel intensities change gradually with maximum variation of 40. Each pixel disappeared during certain time periods as the driving signals take value of \( -1 \). It should be noticed that from \( k = 60 \) to \( k = 70 \), the target disappeared entirely. At the same time, uniform distributed binary clutter of level 0.5% has been added on the background, implying that 0.5% percentage of pixels in the image are noisy. Fig. 5 gives the tracking results on infrared image sequence captured by real airborne infrared camera, where the target experienced obvious appearance change and disappeared temporarily in the frames \( k = 74 \) to \( k = 82 \).

7.3.1 Comparison with fixed observation model IMM-PF tracker

The fixed observation model tracker use the likelihood model as

\[
p(y(k) \mid \Phi(k)) = \prod_{j=1}^{d} N(z_j(k); m_j(k), \sigma_j^2(k))
\]
which is a multivariable Gaussian density with fixed mean and covariance. In Fig. 3, the Gaussian mean of both fixed and adaptive observation models are initialized by the initial values of the driving signals. Figs. 3 (a) and (c) depict the tracking results of IMM-PF trackers with fixed and adaptive observation model respectively. In both figures, true trajectories are represented by real line and estimated trajectories are represented by dashed line with star markers. The 3 times covariance ellipses at certain time instances are also shown in the figures. It is shown in Fig. 3 (a) that after a short period of time, the state covariance diverges rapidly and the target is lost of tracking finally. In fact, as the observation model is fixed, the actual values of target intensity deviate from the mean of Gaussian in the observation model more and more, resulting in a flat distribution of weights among particles and hence a growing covariance ellipses. In contrast, the adaptive observation model tracker, as shown in Fig. 3 (c), can track the target successfully. Fig. 4 (a) depicts the estimate results of the probabilities that the target is present or absent, which are represented by real and dashed lines respectively. During the period from
Fig. 4. Observation model adaptation; each subplot corresponds to one pixel.

(a) Mixing probability estimate. (b) Pixel intensity estimate.

Fig. 5. Single target tracking on real infrared image sequence. The target location is depicted by star (*) and target state covariance is depicted by ellipse; Column 1: Fixed observation model IMM-PF tracker; Column 2: IMM-PDA tracker; Column 3: Adaptive observation model IMM-PF tracker.
When the target disappeared entirely, the target present probability drops quickly and particle weights tend to follow a uniform distribution. This weakens the correction effects of the current measurements on the state and hence the state estimate lies strongly on the dynamic model prediction, as shown in Fig. 3 (c). At the same time, the state covariance ellipse expands and results in a larger particle exploring region. When the target reappears, the estimate of target present probability increases and the measurement correction effect is enhanced. The target is captured again and the covariance ellipse deflates. Fig. 4 (b) depicts the target pixels intensity estimate results, where the real line represents actual intensity and the dashed line represents the estimate. During the periods when the pixels disappear, the actual intensity takes the value of the background at current location. The actual intensity just equals to the driving signals when the target pixel is present and Fig. 4 (b) shows that the target intensity can be estimated well during these periods. In Fig. 5 columns 1 and 3 show the performance of IMM-PF tracker with fixed and adaptive observation model respectively. The observation model was initialized by target patch extracted manually from the first frame of the real image sequence. Fig. 5 shows that fixed observation model tracker lost the target due to target appearance changing. However the adaptive observation model tracker accommodates the target intensity changing and temporarily disappearance and tracks the target successfully.

### 7.3.2 Comparison with IMM-PDA tracker

We also compare our algorithm with the classical IMM-PDA tracker [11]. Probabilistic Data Association (PDA) filter, which use all of the validated measurements with different weights, has been shown to be very effective in tracking target in clutter. The PDA filter, in conjunction with the IMM framework, yields one of the best solutions in the tracking benchmark problem [16] designed to compare the performance of tracking algorithms. The IMM-PDA tracker needs a separate target detection module to provide measurement data. In our simulation, the measured target positions are generated by passing the image frames through a Laplacian filter followed by a thresholder. The pixel intensity driving signals are also shown in Fig. 2.

We test the algorithms on image sequences with different noise levels. For each noise level, 100 Monte Carlo simulations have been ran to calculate the rate of successful tracking. The results are given in Table 1. We find that, if the noise level exceeds certain value (0.4% in our simulation), the IMM-PDA tracker tends to lose the target during the period when the target disappeared and cannot recapture the target when it is present again, resulting in rapidly dropping successful tracking rate. In contrast, the successful tracking rate of our algorithm does not decrease significantly until the noise level exceeds

<table>
<thead>
<tr>
<th>Noise level</th>
<th>0.1%</th>
<th>0.2%</th>
<th>0.3%</th>
<th>0.4%</th>
<th>0.5%</th>
<th>0.6%</th>
<th>0.7%</th>
<th>0.8%</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMM-PDA</td>
<td>0.98</td>
<td>0.94</td>
<td>0.90</td>
<td>0.81</td>
<td>0.64</td>
<td>0.26</td>
<td>0.12</td>
<td>0.07</td>
</tr>
<tr>
<td>Our Algorithm</td>
<td>0.99</td>
<td>0.96</td>
<td>0.94</td>
<td>0.94</td>
<td>0.92</td>
<td>0.92</td>
<td>0.90</td>
<td>0.89</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Noise level</th>
<th>0.9%</th>
<th>1.0%</th>
<th>1.5%</th>
<th>2.0%</th>
<th>2.5%</th>
<th>3.0%</th>
<th>3.5%</th>
<th>4.0%</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMM-PDA</td>
<td>0.02</td>
<td>0.02</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Our Algorithm</td>
<td>0.87</td>
<td>0.84</td>
<td>0.80</td>
<td>0.74</td>
<td>0.63</td>
<td>0.51</td>
<td>0.27</td>
<td>0.15</td>
</tr>
</tbody>
</table>
The robustness against noise interference of our algorithm, as other imaging based tracking algorithms, may be attributed to the more systematic use of target appearance information than that in the detection module of IMM-PDA tracker. However, when the noise density keeps growing after 2%, the successful tracking rate of our algorithm also decreases. The heavy noise increases the possibility that false image patches similar with true target appear in the exploring region. These false image patches are easily misinterpreted by the tracker and leads to loss of tracking.

Fig. 3 (b) shows the result in one simulation of the IMM-PDA tracker under the noise level of 0.5%. In fact, because of the existence of dense clutter, besides the true target measurement, lots of false measurements are usually produced by detection module. However, when the target is present, the true measurement usually presents near the center of validated region and has more weights than others and hence it dominates the measurement update process persistently. This leads to the good performance of IMM-PDA tracker when target is present. When the target disappeared suddenly, the covariance ellipse, or equally the validation region enlarged. The realization of the noise process may result in an unsymmetrical distribution of false measurements in the validation region, making the tracking ellipse deviate from the true trajectory. Even the target appeared again after some time and it has been detected successfully, the true measurement, which is likely to be far from the validation region center, has very little effect on the filter’s measurement updating process dominated by the false measurements located nears the region center. This leads to the final lose of track of IMM-PDA tracker, as shown in Fig. 3 (b). In contrast, our algorithm adapts the observation model automatically and gives more weight to the uniform component in particle likelihood when target disappeared, making the tracker trusts the dynamic model heavily and immune to false measurements in this period. As shown in Fig. 3 (c), when the target appeared again, the corresponding particle will have higher likelihood and attract the tracker back to the true trajectory. In Fig. 5 column 2 shows the tracking result of IMM-PDA tracker on real infrared image sequence. As the noise level is rather low, the IMM-PDA tracker can track the target successfully.

### 7.4 Multiple Targets Tracking

We demonstrate the performance of our multiple targets tracking algorithm (we call it joint tracker in this paper) by compare it with that of the independent tracker, which track multiple targets with several independent single target trackers. Two kinds of single target trackers are used in the comparison, that is, the adaptive observation model IMM-PF tracker and the IMM-PDA tracker. The results are shown in Fig. 6. The true trajectories of the two targets under tracking are depicted by different dashed lines and the estimated trajectories are depicted by real lines with different markers. Note that the two trajectories crossed at time instants $k = 16$ and $k = 60$ respectively. The appearance evolution of target 1 is defined by the driving signals shown in Fig. 2. And the driving signals defining that of target 2 are obtained by setting the values of the original driving signals to $-1$ during the period $k = 15$ to $k = 20$. The difficulty lies in the fact that the appearances of the two targets are identical when they are present and the two trajectories just crossed during the periods when one or both of them disappeared. Specifically, during the period $k = 15$ to $k = 20$ target 2 disappeared while target 1 is still present; during $k = 60$ to $k = 70$ both targets disappeared. As a natural consequence, in these two periods,
the probability distributions of the particles corresponding to the disappearing targets keep diffusing until the targets reappear and then reinforced by the new measurements. However, as the appearances of the two targets are identical, the measurements are easily misinterpreted by the imaging based IMM-PF tracker when one of them disappeared and result in losing tracking of one of targets. As shown in Fig. 6 (a), during period $k = 15$ to $k = 20$, when target 2 disappeared but target 1 is still present, the particles representing target 2 are misenforced by the measurement of target 1, and lead the two trackers to coalesce onto target 1 after their first cross. Similarly, in the case of IMM-PDA tracker, during the period of the first cross of the targets, the validation regions of the two trackers coincide and then the same set of measurements are used in the PDA filter. This leads to the losing of target 2 after $k = 20$, as shown in Fig. 6 (b). In the simulation, when both targets disappeared at $k = 60$ the IMM-PDA trackers diverged due to the interference of noise and do not recaptured the targets after they reappeared. In contrast, the joint tracker precludes the possibility that the two targets are enforced by the new measurement data.
8. CONCLUSION

In this paper, we have presented a new method for tracking small maneuvering target with appearance variation in infrared images sequence. The proposed method has shown superior results compared with the fixed observation model tracker and classical IMM-PDA tracker, in the situation where target experiences large appearance variation and the background clutter is heavy. For multiple target tracking, comparison has been made between our proposed joint tracker and the naïve independent tracker. The results showed that our method can still keep tracking of all targets when one of the two cross-
ing targets disappeared for a short period. Theoretically, the proposed multiple targets tracking algorithm can be used to track any number of targets. However, when the number of targets under tracking increases above a certain level, the number of particles required to model the state distribution can be very large indeed. This curse of dimensionality limits the application of the algorithm to multiple targets tracking seriously. Therefore, a more efficient sampling approach needs to be investigated in future study.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their valuable suggestions for improving the quality of presentation.

REFERENCES

13. J. A. Roecker, “A class of near optimal JPDA algorithms,” *IEEE Transactions on


**Jiu-Qing Wan** (万九卿) received the B.Eng. degree from Hebei University of Technology, Tianjin, China, in 1997, and M.Eng. and Ph.D. degree from Beijing University of Aeronautics and Astronautics, Beijing, China, in 1999 and 2003 respectively, all in electrical engineering. Currently, he is with the School of Automation, Beijing University of Aeronautics and Astronautics. His research interests are signal/image/video processing, statistical inference, target detection, tracking and recognition, and fault diagnosis and prognosis of complex system.

**Xiao-Qing Zhang** (张晓清) received her B.Eng. degree in Electrical Engineering from Anhui University, Hefei, China, in 2006. Currently she is pursuing the M.Eng. in the School of Automation, Beijing University of Aeronautics and Astronautics. Her research interests are digital signal processing and target tracking.

**Jin-Song Yu** (于劲松) received his Ph.D. degree in Automation Science from Beijing University of Aeronautics and Astronautics in 2004. Currently he is an associate professor of School of Automation at Beijing University of Aeronautics and Astronautics, Beijing. His research interests include automatic test system, intelligent diagnosis, prognostics and health management, modeling and simulation.