The Effect of State Dependent Probability of Detection in Multitarget Tracking Applications

A. Gning[a], W T. L. Teacy[b], R. V. Pelapur[c], H. AliAkbarpour[c], K. Palaniappan[c], G. Seetharaman[d], S. J. Julier[a]

[a]Department of Computer Science, University College London, UK
[b]University of Southampton, UK
[c]Department of Computer Science, University of Missouri-Columbia, USA
[d]Air Force Research Lab, USA

ABSTRACT

Through its ability to create situation awareness, multi-target target tracking is an extremely important capability for almost any kind of surveillance and tracking system. Many approaches have been proposed to address its inherent challenges. However, the majority of these approaches make two assumptions: the probability of detection and the clutter rate are constant. However, neither are likely to be true in practice. For example, as the projected size of a target becomes smaller as it moves further from the sensor, the probability of detection will decline. When target detection is carried out using templates, clutter rate will depend on how much the environment resembles the current target of interest.

In this paper, we begin to investigate the impacts on these effects. Using a simulation environment inspired by the challenges of Wide Area Surveillance (WAS), we develop a state dependent formulation for probability of detection and clutter. The impacts of these models are compared in a simulated urban environment populated by multiple vehicles and cursed with occlusions. The results show that accurate modelling the effects of occlusion and degradation in detection, significant improvements in performance can be obtained.

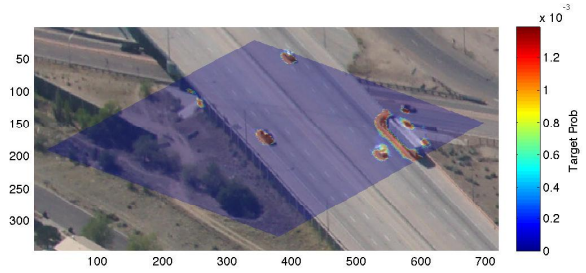
1. INTRODUCTION

Developing situation awareness is vital for almost any kind of military operation.[1] Through understanding the state and nature of the environment, military personnel can plan and respond accordingly. Developing situation awareness in urban environments is particularly challenging due to the clutter, complicated geometry, and limited line-of-sight. One way to improve this is to use Wide Area Surveillance (WAS) using Wide Area Motion Imagery (WAMI).[2] A platform, equipped with a high resolution, wide field of view camera, continuously images the environment. As shown in Figure 1(a), each frame from such a system contains sufficient information to comprehensively monitor large sections of the environment. For example, it possible to identify the locations and trajectories of all vehicles — including those on major roads, minor roads and in car parks — within the order of square kilometres. However, WAMI introduce many challenges. It is an example of massive multi-target tracking in which there can be hundreds or thousands of targets which are in constant motion and interact with one another. The focus of this paper, however, is on the observation process used to extract the targets in the first place.

In general, there is a strong dichotomy between the designers of tracking algorithms and the developers of sensing systems. Tracking algorithm designers typically model sensing systems as black boxes that provide a set of discrete, point-like detections. Some of these detections arise from the target, and are corrupted by noise. Some arise from clutter (either noise in the sensing system itself, or features in the environment which are not a target of interest). However, it is also possible that the target will fail to produce a detection. To model these effects, tracking systems summarise this information in terms of the target likelihood model, a clutter process and a probability of detection. In contrast, sensing systems often internally acquire a dense sampling of the environment with continuous valued measurements. As illustrated in Figure 1(b), a continuous-valued detection or classification algorithm is applied and scores are computed to denote the

Geospatial InfoFusion and Video Analytics IV; and Motion Imagery for ISR and Situational Awareness II,
Matthew F. Pellechia, Kannappan Palaniappan, Shiloh L. Dockstader, Peter J. Doucette, Donnie Self, Eds.,
Proc. of SPIE Vol. 9089, 908905 · © 2014 SPIE · CCC code: 0277-786X/14/$18 · doi: 10.1117/12.2053968

(a) A sample frame from a WAMI dataset.



(b) Score from running a dense detector over a frame.

Figure 1. A typical dataset and the result of applying the detector. The frame is taken with a sufficiently wide field of view and high resolution that a large section of the environment can be imaged in a single frame. The detector has been tuned to detect a particular target vehicle. Although it successfully detects the vehicle in this case, it generates a number of false positives from other vehicles, a truck and infrastructure on the side of the road.

potential existence of targets. Detections are declared if the score meets some criteria (often exceeding a threshold). However, the response of a detection algorithm depends upon a range of factors including the type of target, the relative aspect between the target and the platform, the presence of occlusions, and ambient conditions such as light levels. As a result, the mapping of a sensor measurement to a score value can be nondeterministic. This, in turn, can lead to complicated relationships between a target appearing in the field of view of a sensor, and a detection of that target being generated.

Although some tracking frameworks and algorithms acknowledge the underlying complexities in detecting targets,[3,4,5] most implementations typically use very simple models. The most common ones are that clutter rates and probabilities of detection are time invariant and are constant for all targets in all conditions.

In this paper, we develop a detection-level model to describe state dependency in detection and clutter problems, and explore the impact of this dependency in multi-target tracking algorithms. The structure of this paper is as follows. In Section 2, we describe the motivating problem of WAMI in greater detail and introduce the random set notation used to describe the problem. The process for generating observations is described in Section 3. We introduce or detector-based model of the observation process. Its implementation in a Sequential Monte Carlo SMC formulation of a Probabilistic Hypothesis Density Filter (SMC–PHD Filter) is described in Section 4. The performance of the dynamic model is shown in Section 5 which considers the results in a simulation example of an air platform tracking ground targets. The results show that awareness of occlusion significantly improves the overall performance of the filter. The work is summarised and concluded in Section 6.

2. PROBLEM STATEMENT

2.1 Wide Area Motion Imagery to Conduct Wide Area Surveillance

We wish to track all vehicles within an urban environment. The environment is monitored using an airborne camera sensing array with a high spatial resolution, low frame rate (one to ten frames per second) imaging system. The frame at time step k is $I_k = \{p_{u,v} \in \mathcal{P} | u \in [1, U], v \in [1, V]\}$, where U and V are the width and height of the image, and $\mathcal{P}$ is the set of possible pixels. One frame from a dataset is shown in Figure 1(a). Each frame provides a detailed, high resolution view of a large part of the environment consisting of buildings, roads and vehicles.

At time step k, suppose there are T_k targets. The state of the ith target is $\mathbf{x}_{k,i}$. The state of the whole environment, which consists of the union of the states of each individual vehicle, is the set

$$X_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,T_k}\}. \tag{1}$$

We assume that the environment $\mathbf{e}$, which consists of anything which will impact on the motion of the targets and the ability of the platform to observe them, is known. The position and orientation of the platform, $\overset{*}{\mathbf{x}}_k$, is also assumed to be known.

The advantage of using WAMI is that, in a single frame, it is possible to capture a very large swath of the environment. As a result, it is potentially possible to construct tracks over a large area (and thus develop comprehensive situation awareness). Furthermore, it is possible to track individual targets over many frames. However, tracking in these images is extremely challenging for many reasons. These include the relatively small size of targets, the large number of targets, and changes in appearance due to changes in environmental conditions and the relative attitude between the target and the camera. As a result, any tracking algorithm which is used must robust to these challenges. We propose to address all of these concerns through the use of Mahler's Finite Set framework.[3]

2.2 A Finite Set Statistic Approach to Multi Object Tracking

Mahler argued that the appropriate way to model multi-target problems is to allow X_k to be a random set in which both the *number* of elements in the set, together with the values of those elements, are unknown.[3] Given this formulation, recursive filters can be constructed which have the usual predict–update steps.

The prediction is of the form

$$\Xi_{k+1|k} = \Gamma\left(X_k\right) \cup B. \tag{2}$$

The predicted random set, $\Xi_{k+1|k}$, is formed from the union of two terms. The first term, $\Gamma\left(X_k\right)$, describes the evolution of existing targets. Each target survives with a state dependent probability $P_S\left(\mathbf{x}_{k,i}\right)$. Each target that survives evolves according to the process model, $\pi_{k+1|k}(\mathbf{x}|\mathbf{x}')$. B describes the track birth process in which new targets spontaneously come into being.*

At each time step, the camera on the platform acquires an image. As shown in Figure 1(b), a detection algorithm is run and a set of M_k measurements are extracted. Each measurements consists of the position where it was discovered in the image together with the value of the score function. The set of measurements can be expressed as the set

$$Z_k = \{\mathbf{z}_{k,i}, \ldots, \mathbf{z}_{k,M_k}\} \subset \mathcal{Z}. \tag{3}$$

The measurement model specifies that the observations originate from two sources: the targets and the background,

$$\Upsilon_k = \Upsilon(X_k, \overset{*}{\mathbf{x}}_k, \mathbf{e}) \cup C(\overset{*}{\mathbf{x}}_k, \mathbf{e}). \tag{4}$$

The target detection set, $\Upsilon(X_k, \overset{*}{\mathbf{x}}_k, \mathbf{e})$, is the set of detections which are generated by the targets. It has the structure

$$\Upsilon(X_k, \overset{*}{\mathbf{x}}_k, \mathbf{e}) = \Upsilon(\mathbf{x}_{k,1}, \overset{*}{\mathbf{x}}_k, \mathbf{e}) \cup \ldots \cup \Upsilon(\mathbf{x}_{k,T_k}, \overset{*}{\mathbf{x}}_k, \mathbf{e}) \tag{5}$$

where

$$\Upsilon(\mathbf{x}_{k,i}, \overset{*}{\mathbf{x}}_k, \mathbf{e}) = \begin{cases} \mathbf{z}_i & \text{with probability } p_D(\mathbf{x}_{k,i}, \overset{*}{\mathbf{x}}_k, \mathbf{e}) \\ \emptyset & \text{otherwise} \end{cases} \tag{6}$$

The clutter set, $C(\overset{*}{\mathbf{x}}_k, \mathbf{e})$, is the set of detections which do not originate from the target.

It is important to note that, in general, the likelihood, clutter and probability of detection are all functions of the platform state $\overset{*}{\mathbf{x}}_k$ and the environment $\mathbf{e}$. For example, as the distance between the platform and the target increases, the footprint of the target in the image becomes smaller, and so detection becomes harder.

*For simplicity, we do not consider track spawning in this paper.

2.3 Importance of Observation Model

The proper specification of the observation model is very important. A critical balance exists between the probability of detection and the intensity of the clutter process:

1. If the probability of detection is too high, the filter will view the absence of a detection as very strong evidence that a target no longer exists. As a result, it will tend to eliminate targets too quickly.
2. If the probability of detection is too low, the filter will tend to attribute measurements to clutter and will be slow to react when new targets appear.
3. If the clutter intensity is too high, the filter will tend to attribute detections to clutter and will be slow in declaring the existence of new targets.
4. If the clutter intensity is too low, detections will be viewed as strong evidence of targets and the existence of many spurious tracks can be declared.

The conventional approach is to assume that both the clutter rate and probability of detection are constant, and to manually tune the values until a satisfactory performance is achieved.[6] To this end, some authors have begun to develop algorithms to estimate these quantities. Mahler et al., for example, developed a formulation of the CPHD filter which estimates the probability of detection and clutter rate throughout the detection region of the sensor.[7]

However, as explained above, the performance of detectors is often state dependent. Some authors have recognised this fact and have developed tracking solutions which are able to exploit this. Skoglar, for example, investigated how a state-dependent probability of detection affected the performance of a vehicle tracking application.[4] By modelling probability of detection as a squared Gaussian model on the basis of distance, he demonstrated that significant improvements in performance could be made over the assumption of a fixed probability of detection. However, his work only considered the case of a single vehicle and provided no justification for the form of the probability of detection used. Song presented an extended Kalman Filter-based solution for tracking a single target which used Doppler velocity to filter out clutter.[5] Again, the work demonstrated that the use of the state dependent probability of detection produced improvements in performance, but these were not considered in the context of multi-target tracking.

In this paper, we consider the problem of how to extend multi-target tracking algorithms to handle state dependency. Rather than learn the dependency, however, we develop a model, based on the underlying properties of detectors, which describes how both clutter and a non-binary probability of detection arise.

3. DETECTOR-BASED MODEL OF CLUTTER AND PROBABILITY OF DETECTION

To extract potential targets, a visual classifier computes a continuous-valued score function across the image. This score function is thresholded, and any scores about the threshold are treated as observations. To simplify notation, in this section we suppress the time index k.

3.1 Dense Score Function Evaluation

Consider an image I and let $s(u, v)$ be the the value of the score computed at pixel location (u, v). The score is positively correlated with the likelihood that a target is present at that location within the image: the greater the score, the higher the likelihood. Many different types of classifiers exist. These can be broadly classified as either *class-specific* or *instance-specific*. Class-specific classifiers attempt to identify all objects of the same class in an image. Possibilities include the Histogram of Oriented Gradients,[8] or local colour variations.[9] Instance-specific classifiers attempt to identify the same object from image-to-image without reference to the object's class. One example is correlation-based template matching.[10] The precise details do not matter here; the important thing is that the classifier returns a *dense* set of continuously valued observations, as illustrated in Figure 1(b). More formally, these can be written as the *dense* observation vector $\tilde{Z}\{\tilde{\mathbf{z}}_{(1,1)}, \ldots, \tilde{\mathbf{z}}_{(U,V)}\}$. Figure 1(b) shows an example of the output of a detector which has been tuned to vehicle appearance.[10]

3.2 Characterisation of Classifier-Based Observation Models

To interpret the dense scores, we train a generative Bayesian model to characterise the probability distribution of classifier scores, given that a target is either present or not present. This provides us with two likelihood functions for the classifier score s:

1. The *target detection* likelihood, $L_s(\mathbf{x}) = f(s|\mathbf{x}, \overset{*}{\mathbf{x}}, \mathbf{e})$, which is the conditional probability that the score s was generated by a target with state $\mathbf{x}$; and

2. The *clutter detection* likelihood, $c(s) = c(s|\overset{*}{\mathbf{x}}, \mathbf{e})$, which is the conditional probability that the score s was generated by part of the environment other than a target.

In theory, these likelihood models could be empirically constructed by exhaustively sampling the response of the classifier over all types of targets, relative platform-target poses, and operating conditions. In this work, we assume that a simple model of the classifier is known.

3.3 Target Detection

A detection is declared at (u, v) when $s(u, v)$ exceeds a threshold. Although a fixed threshold could be used we favour the use of a likelihood ratio test,

$$Z = \left\{ \tilde{\mathbf{z}} \in \tilde{Z} \,\middle|\, \frac{L_s(\mathbf{x})}{c(s)} \geq \tau \right\} \tag{7}$$

The size of this set, $|Z| = M$ is therefore controlled by the choice of τ. If τ is low, then the probability of detection will be high. However, more clutter points will be accepted as potential targets as well. On the other hand, high values of τ will result in less clutter, but lowers the probability of detection.

3.4 Target Likelihood

The target likelihood is given by

$$\begin{aligned} L_{\mathbf{z}}(\mathbf{x}) &= f(\mathbf{z}|\mathbf{x}, \overset{*}{\mathbf{x}}, \mathbf{e}) && = f(u, v, s|\mathbf{x}, \overset{*}{\mathbf{x}}, \mathbf{e}) \\ &= f(s|u, v, \mathbf{x}, \overset{*}{\mathbf{x}}, \mathbf{e}) f(u, v|\mathbf{x}, \overset{*}{\mathbf{x}}, \mathbf{e}) && \approx f(s|\mathbf{x}, \overset{*}{\mathbf{x}}, \mathbf{e}) f(u, v|\mathbf{x}, \overset{*}{\mathbf{x}}, \mathbf{e}) \\ &= L_s(\mathbf{x}) L_{u,v}(\mathbf{x}). \end{aligned}$$

We have used the assumption that, conditioned on the actual state of the target, the detected position in the frame and the value of the score function are independent of one another.[†]

3.5 Clutter Intensity

The clutter intensity density is computed from the product of the clutter likelihood $c(\mathbf{z}|\overset{*}{\mathbf{x}}, \mathbf{e})$ and the intensity λ.

Suppose that $P_0^{(c)}$ is the prior probability that an observation is generated by clutter, then by Bayes rule, the posterior probability that an observation, $\mathbf{z}_k$, is a clutter point is given by

$$P^{(c)}(\mathbf{z}_k) = \frac{L_{\mathbf{z}}(\mathbf{x}) P_0^{(c)}}{c(s) P_0^{(c)} + L_{\mathbf{z}}(\mathbf{x}) \left(1 - P_0^{(c)}\right)} \tag{8}$$

The expected number of clutter points in the current observation set is

$$\lambda = \sum_{\mathbf{z}_k \in Z_k} P^{(c)}(\mathbf{z}_k) \tag{9}$$

[†]A case where they would not be independent is, for example, if the target location is in shadow.

Because of the dependency on the platform state, (8) is state dependent.

Assuming a uniform prior over the position of clutter points within the field of view,

$$c(\mathbf{z}|\overset{*}{\mathbf{x}}_k, \mathbf{e}) = \frac{P^{(c)}(\mathbf{z}_k)}{\lambda \cdot E[|\mathcal{S}|]} \tag{10}$$

where $E[|\mathcal{S}|]$ is the size of the expected area visible within the field of view. The clutter intensity is thus

$$\lambda c(\mathbf{z}|\overset{*}{\mathbf{x}}_k, \mathbf{e}) = \frac{P^{(c)}(\mathbf{z}_k)}{E[|\mathcal{S}|]}. \tag{11}$$

3.6 Probability of Detection

The probability of detection is a function of two things. First, the target must lie in the region of the environment visible to the sensor $\mathcal{S}$ which is not obstructed. Second, the response of the score must be sufficiently high that it passes the likelihood ratio test in (8). We write this as

$$p_D(\mathbf{x}_k|\overset{*}{\mathbf{x}}_k, \mathbf{e}) = Pr(\mathbf{x}_k \in \mathcal{S}|\overset{*}{\mathbf{x}}_k, \mathbf{e}) \cdot Pr\left(s > \tau^*|\overset{*}{\mathbf{x}}_k, \mathbf{e}\right). \tag{12}$$

The first term accounts for targets which fall outside of the field of view of the sensor, or are obstructed by parts of the environment. In general, this can account for uncertainty in both the platform and the environment; with our simplifying assumptions here this is a binary function.

To compute the second term, we need to be able to construct a function which will *predict* the scalar response given the target state. We model the uncertainty in this prediction by specifying a distribution on the score.

4. PHD WITH STATE DEPENDENT DETECTION MODELS

We implemented the state dependent probability of detection in an Sequential Monte Carlo implementation of a PHD Filter (SMC–PHD Filter). The reason for doing this is that it is possible, for each particle, to condition on the state of the target platform, which greatly simplifies the calculations.

4.1 Density Representation

The SMC implementation approximates the intensity as a weighted set of N_k particles,

$$D\left(\mathbf{x}_k|Z_{1:k}\right) \approx \sum_{i=1}^{N_k} w_{k|k}^{(i)} \delta\left(\mathbf{x}_{k|k}^{(i)} - \mathbf{x}_k\right), \tag{13}$$

where $Z_{1:k}$ is the sequence of observations up to timestep k, $w_{k|k}^{(i)}$ is the weight of the ith particle, $\mathbf{x}_{k|k}^{(i)}$ is the location of the ith particle, $\delta(\cdot)$ is the vector form of a delta function and

$$\sum_{i=1}^{N_k} w_{k|k}^{(i)} = \eta_{k|k}, \tag{14}$$

is the expected number of targets.

4.2 Algorithm Steps

The SMC–PHD Filter consists of the following steps:

1. **Predict target intensity.** This consists of two steps: predict existing particles forwards, and modelling the spontaneous birth of targets.

   (a) *Predict existing particles forwards.* The process model is applied to each particle $\mathbf{x}_{k-1|k-1}^{(i)}$ to generate a predicted particle $\mathbf{x}_{k|k-1}^{(i)}$. The weights are $w_{k|k-1}^{(i)} = p_s(\mathbf{x}_{k-1}) w_{k-1|k-1}^{(i)}$.

   (b) *Target birth.* The target birth $b(\mathbf{x}_k)$ is modelled using an adaptive birth approach which exploits the measurements from the previous timestep to create a new set of $N_{k,new}$ particles for the current time step.[11] Each particle receives a uniform weight equal to the number of new particles divided by an expected number of targets which appear at each time step denoted ν_k.

   As a result, the predicted intensity is

$$D\left(\mathbf{x}_k | Z_{1:k-1}\right) \approx \sum_{i=1}^{N_{k-1}+N_{k,new}} w_{k|k-1}^{(i)} \delta\left(\mathbf{x}_{k|k-1}^{(i)} - \mathbf{x}_k\right). \tag{15}$$

2. **Compute correction term.** For each measurement $\mathbf{z}_{k,j}$ in Z_k, compute the term

$$\lambda_{k|k-1}\left(\mathbf{z}_{k,j}\right) = \lambda c\left(\mathbf{z}_{k,i} | \overset{*}{\mathbf{x}}_{k-1}, \mathbf{e}\right) + \sum_{i=1}^{N_{k-1}+N_{k,new}} w_{k|k-1}^{(i)} p_D\left(\mathbf{x}_{k|k-1}^{(i)} | \overset{*}{\mathbf{x}}_k, \mathbf{e}\right) L_{\mathbf{z}_{k,j}}\left(\mathbf{x}_{k|k-1}^{(i)} | \overset{*}{\mathbf{x}}_k, \mathbf{e}\right). \tag{16}$$

   Since the probability of detection is not a constant, it must be computed for each particle separately.

3. **Update.** The update re-weights the particles using the PHD pseudo-likelihood. Specifically,

$$w_{k|k}^{(i)} = L_{Z_k}\left(\mathbf{x}_{k|k-1}^{(i)} | \overset{*}{\mathbf{x}}_k, \mathbf{e}\right) w_{k|k-1}^{(i)}, \tag{17}$$

   where

$$L_{Z_k}\left(\mathbf{x}_{k|k-1}^{(i)} | \overset{*}{\mathbf{x}}_k, \mathbf{e}\right) = 1 - p_D\left(\mathbf{x}_{k|k-1}^{(i)} | \overset{*}{\mathbf{x}}_k, \mathbf{e}\right) + \sum_{i=1}^{M_k} \frac{L_{\mathbf{z}_{k,i}}\left(\mathbf{x}_{k|k-1}^{(i)} | \overset{*}{\mathbf{x}}_k, \mathbf{e}\right) p_D\left(\mathbf{x}_{k|k-1}^{(i)} | \overset{*}{\mathbf{x}}_k, \mathbf{e}\right)}{\lambda_{k|k-1}\left(\mathbf{z}_{k,i}\right)}. \tag{18}$$

   Again, the particle-dependent probability of detection must be computed.

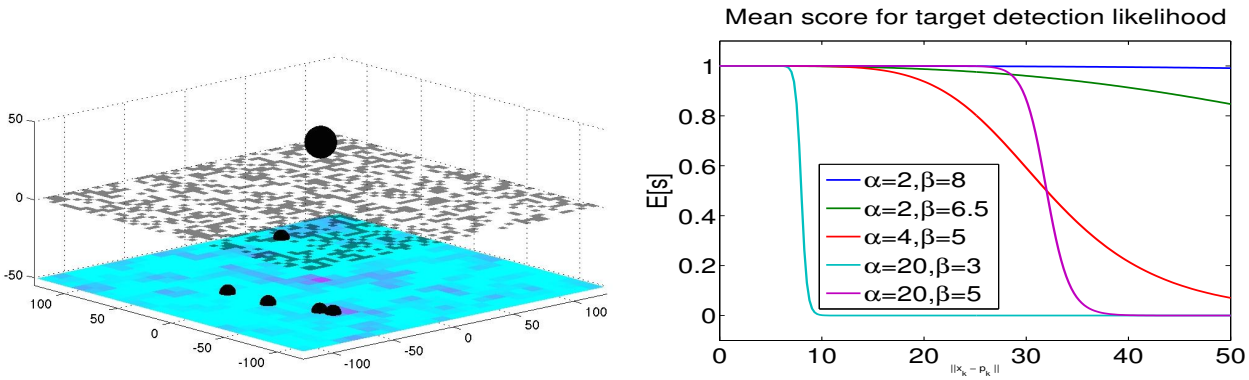
4. **Resample.** As with all SMC implementations, resampling is required to mitigate the effects of particle depletion. The average number of targets is computed from

$$\eta_{k|k} = \sum_{i=1}^{N_k} w_{k|k}^{(i)}. \tag{19}$$

   Stratified resampling is used and the particles are renormalised so that their weights still sum to $\eta_{k|k}$.

5. SIMULATION EXAMPLE

To assess the performance of the algorithm a simplified version of the WAMI problem described in Section 2.1 was developed.



(a) The simulation scenario.

(b) Sigmoid functions used in the target likelihood model.

Figure 2. The simulation scenario. A UAV flies over a large environment and observes targets on the ground. The targets can be obscured by occluding objects. The sensor likelihood is a Gaussian whose mean changes with a sigmoid with distance. When the sigmoid value tends to zero, a target is essentially indistinguishable from clutter.

5.1 Simulation Set-up

The scenario, illustrated in Figure 2(a), was chosen to demonstrate the impact of spatially varying probability of detection in the presence of clutter and occlusions. An aerial platform flies over an area of approximately 400m × 400m. The scenario has a length of 100s. The camera's field of view is modelled as a constant, axis-aligned square whose dimensions are 200m × 200m centered under the current position of the platform.

5.1.1 Platform Trajectory

The platform trajectory simulates a typical trajectory of an aircraft undertaking WAMI by flying in a spiral pattern. The platform state is given by $\overset{*}{\mathbf{x}}_k = \begin{bmatrix} x & y & \delta & \theta \end{bmatrix}_k^\top$, which evolves according to the model

$$\begin{array}{llll} x_{k+1} & = x_k + \Delta T \delta_k \cos(\theta_k) & y_{k+1} & = y_k + \Delta T \delta_k \sin(\theta_k) \\ \delta_{k+1} & = \delta_k + \Delta T u_d u_k & \theta_{k+1} & = \theta_k + \Delta T u_\theta v_k \end{array} \tag{20}$$

The platform's height is a constant $z = 80\text{m}$. $\delta_0 = 5\text{m}$ and $\theta_0 = \pi/20\text{rad}$. u_k and v_k are uniformly distributed random variables in the range $[0, 1]$. $u_d = 2\,\text{m}\,\text{s}^{-2}$ and $u_\delta = \pi/24\,\text{rad}\,\text{s}^{-1}$.

5.1.2 Simulation of Track Trajectories

Each true target moves according to a constrained random walk. The state of the target is given by $\mathbf{x}_k = \begin{bmatrix} x & \dot{x} & y & \dot{y} \end{bmatrix}_k^\top$. This consists of the target position (x, y) and the target velocity, both expressed in Cartesian coordinates. The position of the target is sampled using a 2D multivariate Gaussian with mean $(0\text{m}, 0\text{m})$ and standard deviation $(3\text{m}, 3\text{m})$. The target velocity is sampled with the mean $(1\,\text{m}\,\text{s}^{-1}, 1\,\text{m}\,\text{s}^{-1})$ and corresponding standard deviation $(2\,\text{m}\,\text{s}^{-1}, 2\,\text{m}\,\text{s}^{-1})$.

For each subsequent time step, the process is as follows:

1. With probability p_{death}, an existing target is deleted.
2. With probability p_{birth}, a new target is created. The position of the target is sampled using the platform location as the mean, and the standard deviation $(3\text{m}, 3\text{m})$. The new target's velocity is initialised with mean $(1, 1)$ and standard deviation $(2\,\text{m}\,\text{s}^{-1}, 2\,\text{m}\,\text{s}^{-1})$.

3. An existing, surviving target evolves according to the nearly constant velocity motion model with transitional density $\pi_{k+1|k}(\mathbf{x}|\mathbf{x}') = \mathcal{N}(\mathbf{x}; \mathbf{F}\mathbf{x}', \mathbf{Q})$ where

$$\mathbf{F} = \mathbf{I}_2 \otimes \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}, \qquad \mathbf{Q} = \mathbf{I}_2 \otimes \begin{bmatrix} \frac{\Delta T^3}{3} & \frac{\Delta T^2}{2} \\ \frac{\Delta T^2}{2} & \Delta T \end{bmatrix} \cdot \varpi \tag{21}$$

where $\otimes$ is the Kronecker product, ΔT is the length of each timestep and ϖ is the intensity of process noise. To prevent targets from escaping entirely from the scenario, a further constraint is applied if the distance between the platform and the track becomes too large. Specifically, a control input of the form $(\dot{x}, \dot{y}) = \frac{1}{2}(\dot{x}, \dot{y}) + \gamma((\dot{x}, \dot{y}) - (\delta_k \cos(\theta_k), \delta_k \sin(\theta_k)))$ is applied, where $\gamma = 0.1$. This causes the targets to move back towards the platform.

5.1.3 Observation Model

We model the classifier score, s, as a stochastic function with mean zero for clutter observations, and mean one when a target is observed at close range, but dropping to zero as the distance between the target and camera increases. Therefore,

$$s_i \sim \begin{cases} C_{s_i}\left(\mathbf{x}_{k+1|k}^{(i)}\right) = \mathcal{N}\left(0, \sigma_c^2\right) & \text{clutter} \\ L_{s_i}\left(\mathbf{x}_{k+1|k}^{(i)}\right) = \mathcal{N}\left(m\left(\mathbf{x}_{k+1|k}^{(i)}, \mathbf{p}_k\right), \sigma_t^2\right) & \text{target} \end{cases} \tag{22}$$

where σ_c^2 is the variance for clutter observations, $\mathbf{x}_k$ is the target position and $\mathbf{p}_k$ is the platform position. The mean function, $m(\cdot)$, is a sigmoid defined on the $\log_2$ distance between the target and camera:

$$1 - \frac{1}{1 + \exp[-\alpha(\log_2 ||\mathbf{x}_k - \mathbf{p}_k|| - \beta)]} \tag{23}$$

Figure 2(b) illustrates the response of several sigmoids with distance. The intuition is that the probability of detecting a target is proportional to the number of pixels the target occupies in the image which is (essentially) linearly related to the $\log_2$ distance. The parameters α and β control the distance and rate at which the mean score drops from one to zero. α and β can be learnt from data.

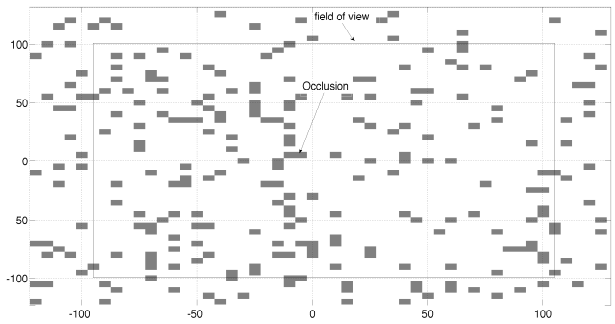


Figure 3. Camera field of view and rectangular occluders.

In addition, the environment has a number of rectangular occluders. These are illustrated in Figure 3.

5.1.4 Probability of Detection

The probability of detection is computed from (12). This requires the solution of two terms: the probability that the target is visible, and the probability that the target will generate a response which will be detected.

Given the assumption that both $\overset{*}{\mathbf{x}}_k$ and $\mathbf{e}$ are known perfectly, we perform a simple ray test to determine if the target lies within an unoccluded detection region of the sensor. The probability is a binary value which either takes the value 0 (not visible) or 1 (visible).

We must compute the probability that

$$\frac{L_s(\mathbf{x}_k)}{c(s)} \geq \tau. \tag{24}$$

Given our assumption that both $L_s(\mathbf{x}_k)$ and $c(s)$ are Gaussian, we compute the smallest value of s^* which will trigger a detection as follows,

$$\begin{aligned} \tau &< L_{s^*}(\mathbf{x}_k)/c(s^*) && (25)\\ \tau &< \mathcal{N}(s^*; m(\mathbf{x}_k, \mathbf{p}_k), \sigma_t^2)/\mathcal{N}(s^*; 0, \sigma_c^2) && (26)\\ \ln \tau &< \frac{1}{2}\ln\left(\frac{\sigma_c^2}{\sigma_t^2}\right) + \frac{(s^*)^2}{2\sigma_c^2} - \frac{(s^* - m(\mathbf{x}_k, \mathbf{p}_k))^2}{2\sigma_t^2} && (27) \end{aligned}$$

The quadratic can be solved to determine the value of s^*. We then need to compute the probability $L_s(\mathbf{x}_k) \geq s^*$. Since $L_s(\mathbf{x}_k)$, this value can be computed from the error function.

Variable	Value	Description
p_{birth}	0.2	birth rate of ground moving targets
p_{death}	0.05	death rate
$\sigma_{v_x} = \sigma_{v_y}$	$0.7\mathrm{m\,s^{-1}}$	Evolution model, nearly constant velocity model
N	2000	number of particles
N_b	1500	number of new born particles
τ	0.85	threshold to decide if measurement originated from a target

Table 1. The parameters used in the simulation.

5.2 Results

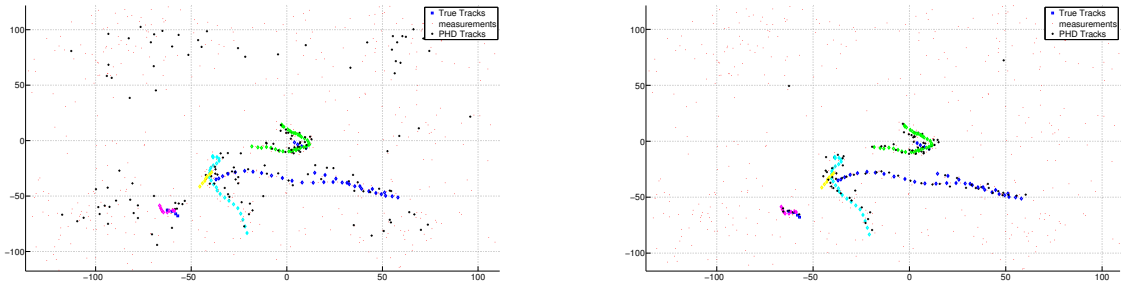
In this section, we consider two scenarios and compare the performance of two filters. The two scenarios, listed in Table 2, can be broadly described as "easy" and "hard". In the easy scenario, the clutter rate is relatively low and the response of the detector to the target is relatively flat over a large distance. In the hard scenario, the clutter rate is much higher, and the detection probability drops much more strongly with distance. The two filters are:

- **PHD.** This is a conventional PHD filter. The clutter rate and probability of detection are fixed constants. To compute the values, both the clutter rate and the probability of detection correspond to an averaged value over the image pixels and the varying distance between the platform and the ground.
- **DynPHD.** This is the PHD filter with the dynamic probability of detection and clutter rates.

Figures 4(a) and 4(b) show a single frame from the performance of each filter in an illustrative simulation run where the sigmoid parameters are set to $\alpha = 2$ and $\beta = 7$ (corresponding to an average probability of detection equal to 0.87 and a clutter rate equal to 23). As can be seen, the regular PHD filter maintains many more tracks (crosses) than ground truth. DynPHD exhibits far fewer spurious tracks.

Scenarios	probability of detection	clutter rate
sigmoid parameters $\alpha = 2$, $\beta = 8$	0.90	3
sigmoid parameters $\alpha = 2$, $\beta = 6.5$	0.80	80

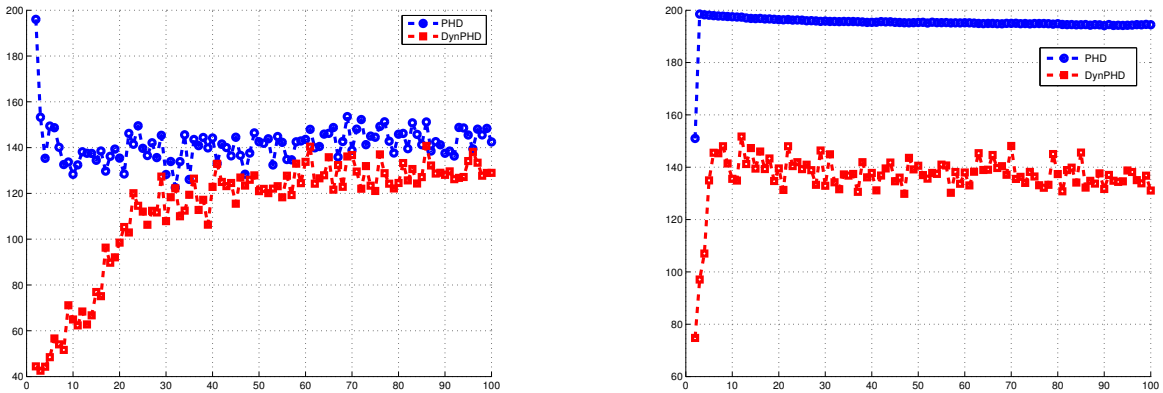
Table 2. The two scenarios and parameters used for the PHD.



(a) Conventional PHD filter. (b) DynPHD filter.

Figure 4. Illustration of trajectories generated by the different PHD filters in comparison to the ground truth. The dots represent the measurements, the circles trajectories of the true targets and the plus PHD tracks. Note that the conventional filter generates many spurious tracks.

To quantitatively compare the performance of the two filters, we used the OSPA[12] metric, which provides a mechanism to evaluate both cardinality and position errors in terms of a distance measure[‡]. Figure 5(a) shows the results from the two filters for 100 Monte Carlo runs for the first scenario. This clearly shows the improvement in performance when considering occlusions only (high p_D and low clutter as can be seen in Table 2. Figure 5(b) shows the results when the probability of detection falls more sharply with range. As can be seen, the conventional PHD filter performs much worse, whereas the performance of DynPHD has only degraded very slightly. While both filters suffer from track discontinuity mainly due to limited field of view and occlusion, the DynPHD offers both better cardinality estimation and accuracy. The DynPHD filter is more robust to clutters and more crucially, the pseudolikelihood is improved and yields to a better Monte Carlo approximation of the posterior intensities.



(a) Average probability of detection of 0.90. (b) Average probability of detection of 0.80.

Figure 5. OSPA metrics averaged over100 Monte Carlo runs for both the static and dynamic PHD filters. The OSPA metric is computed with $c = 200$ and $p = 4$.

6. SUMMARY AND CONCLUSIONS

In this paper, we have considered the problem of state dependency in multitarget tracking using PHD filters. We derived a framework for the probability of detection and clutter rates directly from the properties of the sensor. We have shown that, in a simple simulation scenario, the algorithm can outperform tracking systems which use fixed values.

[‡]In the results reported in this paper the cutoff parameter is set to $200m$ while the OSPA parameter is set to 1.

Future work is exploring how the algorithms can be applied in actual operational systems, using machine learning algorithms to learn the likelihood functions and template-based matching algorithms.[10,13,14]

ACKNOWLEDGMENTS

This work was partially sponsored by the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA8655–12–1–2142 and AFRL grants FA8750–14–C–0043, FA8750–14–2–0072 and FA8750-14–C–0047.

The U.S Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright notation thereon.

REFERENCES

[1] Endsley, M. R., "Theoretical Underpinnings of Situation Awareness: A Critical Review," in [*Situation Awareness Analysis and Measurement*], Endsley, M. R. and Garland, D. J., eds., ch. 1, 3–28, Taylor & Francis (2000).

[2] Palaniappan, K., Rao, R., and Seetharaman, G., "Wide-Area Persistent Airborne Video: Architecture and Challenges," in [*Distributed Video Sensor Networks: Research Challenges and Future Directions*], Bhanu, B., Ravishankar, C. V., Roy-Chowdhury, A. K., Aghajan, H., and Terzopoulos, D., eds., 349–371, Springer (2011).

[3] Mahler, R. P. S., [*Statistical Multisource–Multitarget Information Fusion*], Artech House (2007).

[4] Skoglar, P., Orguner, U., Törnqvist, D., and Gustafsson, F., "Road Target Tracking with an Approximative Rao-Blackwellized Particle Filter," in [*Proceedings of FUSION 2009*], 17–24 (6–9 July 2009).

[5] Song, T. L., Musicki, D., and Sol, K. D., "Target Tracking With Target State Dependent Detection," *IEEE Transactions on Signal* **59**, 1063–1074 (March 2011).

[6] Wol, "Advanced Target Tracking Techniques," Tech. Rep. RTO-EN-SET-086, NATO (2007).

[7] Mahler, R. P. S., Vo, B. T., and Vo, B. N., "CPHD Filtering in Unknown Clutter Rate and Detection Profile," *IEEE Transactions on Signal Processing* **59**, 3497–3513 (Aug 2011).

[8] Dalal, N. and Triggs, B., "Histograms of oriented gradients for human detection," in [*Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*], **1**, 886–893 vol. 1 (June 2005).

[9] Hinz, S., Lenhart, D., and Leitloff, J., "Detection and Tracking of Vehicles in Low Framerate Aerial Image Sequences," in [*ISPRS Archives*], *W51* **XXXVI-1** (2007).

[10] Pelapur, R., Palaniappan, K., and Seetharaman, G., "Robust Orientation and Appearance Adaptation for Wide-Area Large Format Video Object Tracking," in [*9th International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*], 337–342 (18–21 September 2012).

[11] Ristic, B., Clark, D., and Vo, B.-N., "Improved SMC implementation of the PHD filter," in [*Proc. of the 13th International Conference on Information Fusion*], (July 2010).

[12] Ristic, B., Vo, B.-N., Clark, D., and Vo, B.-T., "A Metric for Performance Evaluation of Multi-Target Tracking Algorithms," *IEEE Transactions on Signal Processing* **59**, 3452–3457 (July 2011).

[13] Pelapur, R., Candemir, S., Bunyak, F., Poostchi, M., Seetharaman, G., and Palaniappan, K., "Persistent Target Tracking Using Likelihood Fusion in Wide-Area and Full Motion Video Sequences," in [*Proceedings of FUSION 2012*], (July 2012).

[14] Ersoy, I., Palaniappan, K., Rao, R., and Seetharaman, G., "Tracking in persistent wide-area motion imagery," in [*Proc. SPIE Conf. Geospatial InfoFusion II (Defense, Security and Sensing: Sensor Data and Information Exploitation)*], **8396** (2012).