Video Tracking Using Dual-Tree Wavelet Polar Matching and Rao-Blackwellised Particle Filter

Sze Kim Pang, James D. B. Nelson, Simon J. Godsill, Nick Kingsbury

***Abstract*—A video tracking application using the Dual-Tree polar matching algorithm is described. Dynamic and observation models are developed in a probabilistic setting and a study of the empirical probability distribution of the polar matching output is given. Both the visible and occluded target statistics are modelled by a Beta distribution. This is incorporated into a Track-Before-Detect (TBD) solution for the overall observation likelihood of each video frame and provides a principled derivation of the observation likelihood. Due to the non-linear nature of the problem, we design a Rao-Blackwellised Particle Filter (RBPF) for the sequential inference. Computer simulations demonstrate the ability of the algorithm to track a simulated video moving target in an urban environment with complete and partial occlusions.**

***Index Terms*—Dual-Tree Wavelet,Polar Matching,Video Tracking, Rao-Blackwellised Particle Filter, Track-Before-Detect.**

I. Introduction

Detection and tracking of a known target in video sequences is a common and important problem in image processing. In this paper, we focus on the scenario of an unmanned air vehicle (UAV) platform based image sensor as it attempts to track a ground vehicle traversing a cluttered urban environment. The objective is to provide a good estimate of the position and velocity of the vehicle in grid coordinates, and be robust against temporary occlusions.

As the location of the UAV and target vary, and as the bearing and azimuth of the sensor change, the image of the target will appear to shift and rotate, and possibly change in scale. In this context, it therefore makes sense that any successful detection method must have robustness or invariance to spatial shifts, rotations, and some scale variations.

With this in mind, the descriptor and matching technique afforded by approximate rotation-invariant polar matching with dual-tree complex wavelet transforms (DTCWT) recently developed by Kingsbury in 2006 [1] is adapted here to the task of detection. The output of the polar matching method gives a detection confidence (or likelihood value) of the target of interest for a specific position and orientation within the video frame.

Many approaches have been proposed to tackle the problem of target tracking. These range from Kalman filter and its non-linear extensions to JPDAF trackers [2][3]. With the parallel advances in modern computational power and the developments in optimal non-linear techniques such as particle filters [4][5] and Markov Chain Monte Carlo (MCMC) [6][7], it is now possible to consider the exploitation of other information (such as non-linear measurement processes) which can potentially offer better performances.

The detection output of the polar matching method can be fed into a tracking filter to provide smooth estimates of the target's position. However, an optimal linear filter such as the Kalman filter may not work as well in this scenario. One reason is due to the non-linear measurement process of the imaging sensor and the polar matching method. Another reason is that the posterior distribution is likely to be multi-modal due to the nature of the video data. To overcome these issues, we make use of optimal non-linear filtering techniques based on sequential Monte Carlo methods [4][5], also known as particle filters, to perform the tracking.

The paper is organised as follows. Section II presents rotation-invariant dual-tree complex wavelet polar matching. Section III describe the probabilistic state-space model. Section IV and V describe the dynamic models and observation model respectively. Section VI describes the particle filter algorithm, and Section VII describes the Rao-Blackwellised Particle Filter. Simulation results are discussed in Section VIII, followed by our conclusions in Section IX.

II. Polar matching

Extending his work on the shift-invariant dual-tree complex wavelet transform [8], Kingsbury recently introduced the rotation-invariant polar matching method [1]. Owing to low redundancy, the DTCWT descriptor is more efficient than the existing popular scale- and rotation-invariant methods of SIFT [9] and Simoncelli's steerable pyramids [10]. It is adapted here to provide image matching between a small template and a larger image rather than matching keypoints of two similarly sized images, as previously reported.

Unlike regular wavelet constructions, the DTCWT is shift invariant and, at each scale, decomposes an image into six complex, rather than three real, directional subbands. These properties are exploited by the polar matching correlator. Kingsbury's method proceeds by firstly computing the DTCWT coefficients of a template. The complex wavelet coefficients at each of the six directional subbands at the centre point are stored together with their complex conjugates. Coefficients are also taken around one or more circles, about the centre point, at 30 degree increments from different directional subbands and at multiple scales. As Figure 1 illustrates, the coefficients are then arranged into a polar matching matrix

Sze Kim Pang, James D. B. Nelson, Simon J. Godsill and Nick Kingsbury are with the Signal Processing and Communications Laboratory of the Cambridge University Engineering Department, Trumpington Street, Cambridge CB2 1PZ, UK. (emails: {skp31, jdbn2, sjg, ngk}@cam.ac.uk)

Manuscript received May 1, 2008; revised March 31, 2008.

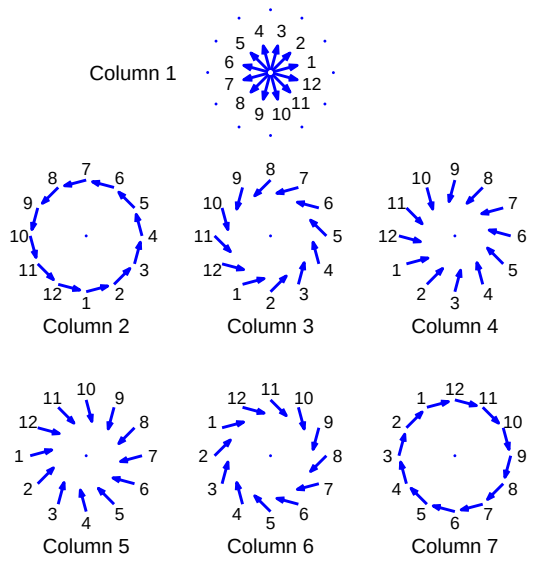


Fig. 1. Locations and orientations of the DTCWT coefficients. Each orientation describes a coefficient, or conjugate, of one of the six subbands. The column numbers indicate the column of the P-matrix in which the coefficients are placed, and the numbers displayed around each circle indicate the row of the P-matrix that each coefficient is placed. Taken from [1].

(P-matrix) such that a rotation of $k \times 30°$ in the original image will produce a vertical shift by k rows in the P-matrix. Consider two images, one a $n \times 30°$ rotated version of the other; then a sum of column-wise correlations between the two corresponding P-matrices will result in a response curve, with respect to relative rotation angle, and a maximum at location n.

However, the rotational sensitivity can be increased to $7.5°$ by careful band-limited interpolation. This is achieved by performing the correlation as a product in the Fourier domain and by zero padding. Care should be taken here over the phase rotations of the complex coefficients. The first column of a P-matrix, formed about the centre of a single step edge will vary slowly as the the edge is rotated. Columns 2 and 7 will vary in phase quicker, 3 and 6 quicker still and 4 and 5 quickest of all. Hence, the zero padding must be placed according to the P-matrix column number. Coefficients obtained from other scales, or colour bands, can be added by appending them as extra columns to the P-matrix. Hence, this polar matching technique takes the property of shift invariance from the DTCWT, and approximate rotation invariance from the P-matrix construction. As explained further in [1], the matching operator is rotation invariant in the sense that we only need to construct a single P-matrix for the template and the correlation score output of all the required angles of orientation can be obtained via a single operation using polar matching. By the careful radial sampling and ordering of the directional DTCWT subband coefficients, polar matching transforms rotations in the object into shifts in the feature vectors. We can contrast this with a naive template correlation matching, where the template is rotated by each angle before the correlation score is calculated. In Figure 2, we plot the result of correlating the original template with rotated copies. The maximum of each curve indicates the degree of match

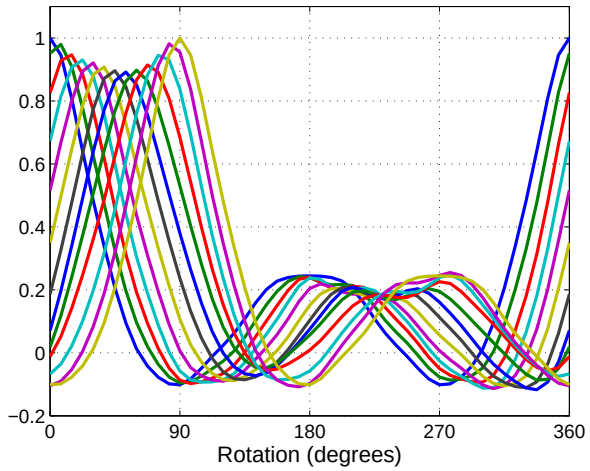


Fig. 2. Correlation results for the template shown in Figure 3: each curve represents the polar matching output obtained by correlating the template with rotated versions of itself, in $5°$ increments from $0°$ to $90°$. For each curve, the output is a response with respect to 48 angles in $7.5°$ increments.

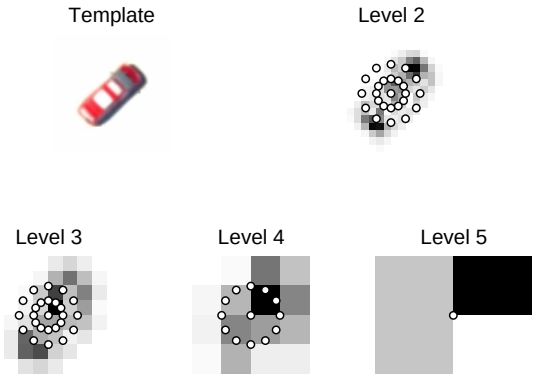


Fig. 3. DTCWT coefficients of a template. The white dots indicate the locations of the coefficients that are used in the P-matrix template. Four different scales, or levels, are used. Only one of one of the 6 subbands is shown.

and its location correctly gives the object orientation.

Define the column vector $\mathbf{h} \in \mathbb{C}^n$ as the concatenation of the Fourier transform of the P-matrix columns. Likewise, let $\mathbf{f} = \mathbf{f}(\mathbf{x})\colon \mathbb{R}^2 \mapsto \mathbb{C}^n$ be the concatenation of Fourier coefficients of the P-matrix of a test image taken at the point $\mathbf{x}$. Together, the inverse Fourier transform and zero padding can be described as a block diagonal matrix operator $\mathbf{W}$. Then, the polar matching operation at the angle θ can be written as

$$\big(\mathbf{h} \star \mathbf{f}\big)(\theta) = \frac{\Re\left(\mathbf{h}^H \mathbf{W}(\theta)\mathbf{f}\right)}{\|\mathbf{h}\|_2 \, \|\mathbf{f}\|_2}, \tag{1}$$

where the superscript ^{H} denotes conjugate, or Hermitian, transpose, and the real component $\Re$ is taken to return the correlation intensity. In practice, the feature vectors are normalised to have unit ℓ_2-norm, the numerator of (1) is merely a weighted dot product and we have that

$$\big(\mathbf{h} \star \mathbf{f}\big)(\theta) = \Re\left(\sum_{i \in \mathbb{Z}} w_i(\theta) \overline{h_i} f_i\right).$$

The target location and orientation can be found by computing $\arg\max_{\mathbf{x},\theta} \big(\mathbf{h} \star \mathbf{f}(\mathbf{x})\big)(\theta)$. Colour images, and coefficients from other scales, are dealt with by simply concatenating the P-matrix Fourier coefficients of each colour channel into the vectors $\mathbf{h}$ and $\mathbf{f}$. In this work, we use the RGB colour bands at for each of the scale levels and sampling strategies shown in Figure 3.

In summary, polar matching provides a correlation score between -1 and 1 for a template in a specific position (X_{Image}, Y_{Image}) and orientation θ within a larger video frame. In the following, the polar matching function is referred to as Polar$(X_{Image}, Y_{Image}, \theta)$. For the video tracking application, the template will be provided as a P-matrix, taken directly from an image of the target vehicle.

III. Bayesian Filtering

We first develop a probabilistic framework for the single target video tracking problem. We are interested in the target's position (x, y) and velocity $(\dot{x}, \dot{y})$ in grid coordinates, as well as the orientation of the image template, θ, with respect to each video frame. Furthermore, the target of interest may be fully or partially occluded due to buildings or other visual occlusions such as smoke. Hence, we introduce a visibility variable V to model this. The joint state at time t is given by $S_t = [x_t \ \dot{x}_t \ y_t \ \dot{y}_t \ \theta_t \ V_t]$, and will be inferred from the video sequences.

Assuming a Markovian state transition, the standard state update and prediction equations are given by

$$p(S_t|Z_{1:t}) = \frac{p(Z_t|S_t)p(S_t|Z_{1:t-1})}{p(Z_t|Z_{1:t-1})} \tag{2}$$

$$p(S_t|Z_{1:t-1}) = \int p(S_t|S_{t-1})p(S_{t-1}|Z_{1:t-1})\,\mathrm{d}S_{t-1} \tag{3}$$

with $Z_{1:t} = [Z_1 \ \cdots \ Z_m \ \cdots Z_t]$ and where Z_m denotes all of the observations collected at time m.

IV. Dynamical Models

We choose to write the transition probability model as

$$p(S_t|S_{t-1}) = p(X_t|X_{t-1})p(\theta_t|\theta_{t-1})p(V_t|V_{t-1}) \tag{4}$$

where $X_t = [x_t \ \dot{x}_t \ y_t \ \dot{y}_t]$. The variables X_t, θ_t and V_t are modelled to be independent of each other. It is also possible to make the orientation θ_t partially dependent on the target's position and velocity. Here, for simplicity, we use the independent model.

For the target dynamic, we use the discrete time equivalent of the near constant velocity model [11]. This is given by

$$X_t = FX_{t-1} + w_t \tag{5}$$

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

where w_t is a Gaussian noise with covariance Q_w given by

$$Q_w = \begin{bmatrix} \frac{T^3}{3}q_x & \frac{T^2}{2}q_x & 0 & 0 \\ \frac{T^2}{2}q_x & Tq_x & 0 & 0 \\ 0 & 0 & \frac{T^3}{3}q_y & \frac{T^2}{2}q_y \\ 0 & 0 & \frac{T^2}{2}q_y & Tq_y \end{bmatrix} \tag{7}$$

with q_x and q_y being the variance of the driving noise of the dynamical process. Hence,

$$X_t|X_{t-1} \sim \mathrm{N}(FX_{t-1}, Q_w) \tag{8}$$

The orientation θ_t models the changes in the orientation of the target at time t. This allows us to track the vehicle as it rotates in plane within the image. This is modelled as a random walk:

$$\theta_t|\theta_{t-1} \sim \mathrm{N}(\theta_{t-1}, S_\theta) \tag{9}$$

where S_θ is the variance of the driving noise. The visibility variable V_t determines whether the target is visible or if it is temporarily obscured by smoke or walls. This affects the way the likelihood term $p(Z_t|X_t)$ behaves. The target's visibility variable will be modelled as a discrete Markov chain,

$$p(V_t = 0|V_{t-1} = 1) = P_{NV} \tag{10}$$

$$p(V_t = 1|V_{t-1} = 0) = P_V \tag{11}$$

V. Observation Model using Polar Matching

In order to construct a principled derivation of the observation model $p(Z_t|X_t, \theta_t, V_t)$, we first imagine each video frame as a discretised grid with N_{grid} possible positions. (Actual discretisation of the video frame is not important here, as the result will be the same.) We assume that the target of interest will always be within the video frame, but that it may or may not be visible. When it is visible, it can occupy only one of the positions in the grid. This approach is similar to the Track-Before-Detect (TBD) type of observation model that are found in literature [12][13].

The grid measurements are modelled as conditionally independent given the states X_t, θ_t and V_t.

$$p(Z_t|X_t, \theta_t, V_t) = \prod_{q=1}^{N_{grid}} p(Z_{t,q}|X_t, \theta_t, V_t) \tag{12}$$

Suppose that the target is visible, i.e. $V_t = 1$, and that it lies in the r^{th} grid position

$$\begin{aligned} & p(Z_t|X_t, \theta_t, V_t = 1) \\ = \ & p(Z_{t,r}|X_t, \theta_t, V_t = 1) \prod_{q=1,\ q\neq r}^{N_{grid}} p(Z_{t,q}|X_t, \theta_t, V_t = 0) \\ \propto \ & \frac{p(Z_{t,r}|X_t, \theta_t, V_t = 1)}{p(Z_{t,r}|X_t, \theta_t, V_t = 0)} \end{aligned} \tag{13}$$

Similarly, if the target is occluded at the r^{th} grid position, i.e. $V_t = 0$, we can write

$$\begin{aligned} & p(Z_t|X_t, \theta_t, V_t = 0) \\ = \ & p(Z_{t,r}|X_t, \theta_t, V_t = 0) \prod_{q=1,\ q\neq r}^{N_{grid}} p(Z_{t,q}|X_t, \theta_t, V_t = 0) \\ \propto \ & 1 \end{aligned} \tag{14}$$

The above provides a simple framework for the detection of a single target using a template based detector. What remains

is to define $p(Z_{t,r}|X_t, \theta_t, V_t)$, which is the probability of observation at the r^{th} grid position using the polar matching output with a specific template. $Z_{r,t}$ is then given by

$$Z_{r,t} = \text{Polar}(X_{Image}, Y_{Image}, \theta) \tag{15}$$

where (X_{Image}, Y_{Image}) can be obtained by a non-linear transformation $H(\cdot)$ which maps the real world coordinate (X, Y) to the image plane. In this paper, we assumed that the sensor's position and orientation is known, i.e. $H(\cdot)$ is known.

We study the histogram of the polar matching output using 100 video frames. For each frame, we will sample random positions in the video frame for both visible and occluded target positions. We can see this sampling done in one of the particular video frame in Figure 4. For simplicity, we treat negative correlation numbers from the polar matching output as zero correlation, i.e. $Z_{r,t} \in [0, 1]$. Figure 5 shows the combined histograms for 100 video frames.

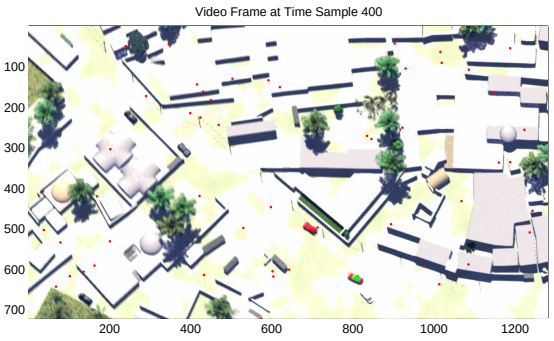


Fig. 4. Red and green dots shows the sampling for the occluded target and visible target histogram respectively.

In probability theory and statistics, the Beta distribution is a family of continuous probability distributions defined on the interval $[0, 1]$ parameterized by two positive shape parameters, typically denoted by α and β. The probability density function is given by:

$$\begin{aligned} \text{Beta}(s|\alpha, \beta) &= \frac{(s)^{\alpha-1}(1-s)^{\beta-1}}{\int_0^1 (s)^{\alpha-1}(1-s)^{\beta-1} \mathrm{d}s} \\ &= \frac{1}{B(\alpha, \beta)}(s)^{\alpha-1}(1-s)^{\beta-1} \end{aligned} \tag{16}$$

where $B(\alpha, \beta)$ is the Beta function. We use the Beta distribution to model both the visible and occluded observation histograms. This is shown in Figure 6. Substituting the Beta distribution into (13) gives

$$\begin{aligned} p(Z_t|X_t, \theta_t, V_t = 1) &\propto \frac{\text{Beta}(Z_{t,r}|5, 7)}{\text{Beta}(Z_{t,r}|1, 7)} \\ &= \frac{B(1, 7)}{B(5, 7)}(Z_{r,t})^4 \end{aligned} \tag{17}$$

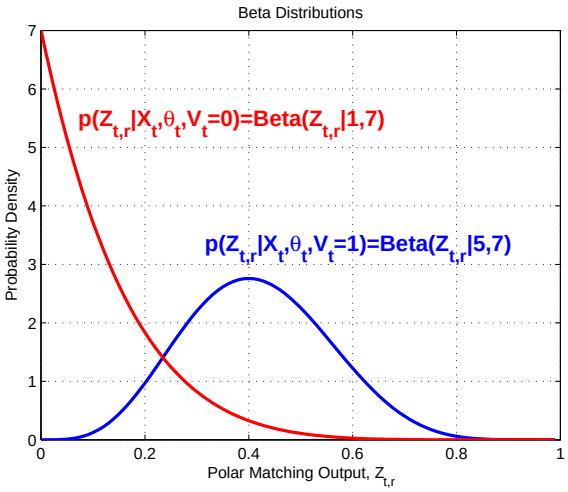


Fig. 6. This figure shows the Beta distributions used to fit the visible and occluded polar matching observation output

VI. Particle Filter Algorithm

The filtering distribution of the dynamical and observation probability model above is complex and non-linear. Sequential Monte Carlo methods such as particle filters [12] can be used to do the inference. A particle filter represents the required posterior density function by a set of random samples (or particles) with associated weights $\{S_{t,p}, w_{t,p}\}_{p=1}^{N_p}$. These particles are then propagated through time to give predictions of the posterior distribution function at future time steps. As the number of samples becomes very large, this monte-carlo characterization becomes an equivalent representation to the usual functional description of the posterior density function. The posterior filtered density at time t is approximated by

$$p(S_t|Z_{1:t}) \approx \sum_{p=1}^{N_p} w_{t,p}\delta(S_t - S_{t,p}) \tag{18}$$

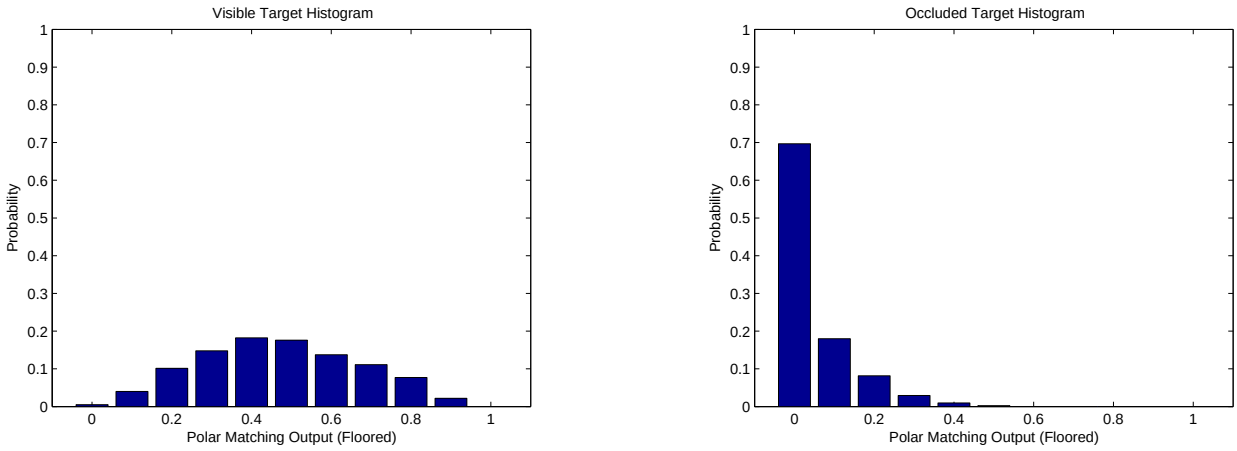
where $Z_{1:t} = [Z_1 \ \cdots \ Z_m \ \cdots Z_t]$ are the observations and the weight $w_{t,p}$, of the particle p, is updated according to

$$w_{t,p} = w_{t-1,p} \times \frac{p(Z_t|S_{t,p})p(S_{t,p}|S_{t-1,p})}{q(S_{t,p}|S_{t-1,p}, Z_t)} \tag{19}$$

The choice of the importance density $q(S_{t,p}|S_{t-1,p}, Z_t)$ is one of the most critical issues in particle filter design. It can be shown that the optimal importance density (in the sense of minimizing the variance of the importance weights), conditioned upon $S_{t-1,p}$ and Z_t is $p(S_{t,p}|S_{t-1,p}, Z_t)$ [5]. There are other suboptimal choices. For example, a popular choice is to use the prior model density $p(S_{t,p}|S_{t-1,p})$. When substituted into Equation (19), we obtain

$$w_{t,p} = w_{t-1,p} \times p(Z_t|S_{t,p}) \tag{20}$$

The simple and general algorithm above forms the basis of most particle filters. However, the algorithm above will result in the variance of the importance weights increasing over time [5]. This will adversely affect the accuracy and lead to the degeneracy problem where, after a certain number



(a) Visible Target Histogram

(b) Occluded Target Histogram

Fig. 5. These figures show the visible and occluded target histogram for 100 video frames.

of recursive steps, all but one particle will have negligible normalized weights. This will result in a large computational effort devoted to updating particles whose contribution to the approximation of $p(S_t|Z_{1:t})$ is almost zero. A practical measure of the degeneracy of the particle weights is the effective sample size N_{eff} introduced in [14]:

$$\hat{N}_{eff} = \left(\sum_{p=1}^{N_p} {w_{t,p}}^2 \right)^{-1} \tag{21}$$

It is easy to see that $1 \leq N_{eff} \leq N_p$. A small N_{eff} indicates a degeneracy problem. When this occurs (for example when N_{eff} drops below some threshold N_{thr}), a step called resampling [4] has to be performed. Resampling eliminates samples with low weights and multiplies samples with high importance weights. This involves mapping a random measure $\{S_{t,p}, w_{t,p}\}_{p=1}^{N_p}$ into a random measure $\{S_{t,p}, \frac{1}{N_p}\}_{p=1}^{N_p}$ with uniform weights.

There are several methods available when implementing the remapping step. The first introduction of resampling [4] uses random sampling of the particles based on their weights. However, a complete random selection is not necessary and it increases the Monte Carlo variation of the particles. Other methods such as stratified sampling [15] and residual sampling [16] may be applied. Systematic Resampling [15] is another efficient method. It is simple to implement, it has order N_p computational complexity and it minimizes the MC variation.

In this paper, we make use of the Sampling-Importance Sampling-Resampling (SIR) filter to perform the inference. We use the prior $p(S_t|S_{t-1})$ as the importance function. For the resampling step, we use Systematic Resampling.

VII. Rao-Blackwellised Particle Filter (RBPF)

Rao-Blackwellisation [17][5] in particle filter is a variance reduction technique that exploits the structure of the probabilistic model. The key idea is that the state space S_t can be partitioned into two parts, $[S_t^L \ S_t^N]$, such that the posterior distribution of the non-linear part S_t^N can be simulated using particles, and the posterior distribution of the linear part S_t^L can be updated analytically given S_t^N and some sufficient statistics. Compared to a normal particle filter that simulates the entire state space S_t using particles, the RBPF will be more efficient, and has lower variance due to exact simulation of the conditionally linear part of the state space.

For the video tracking model using Eq. 5 and 12, it is possible to Rao-Blackwellise the state space with $S_t^N = \{X_t, \theta_t\}$ and $S_t^L = \{V_t\}$.

The derivation of the RBPF can be found in Appendix A. The sufficient statistic $p(V_t|S_{1:t}^N, Z_{1:t})$ is updated over time. The weights of the RBPF can then be written as

$$w_{t,p} = w_{t-1,p} \times \frac{p(Z_t|Z_{1:t-1}, S_{1:t}^N) p(S_{t,p}^N|S_{t-1,p}^N)}{q(S_{t,p}^N|S_{t-1,p}^N, Z_t)} \tag{22}$$

where

$$p(Z_t|Z_{1:t-1}, S_{1:t}^N) = \sum_{V_t \in \{1,0\}} p(Z_t|S_t^N, V_t) p(V_t|S_{1:t}^N, Z_{1:t-1}) \tag{23}$$

and

$$p(V_t|S_{1:t}^N, Z_{1:t-1}) = \sum_{V_{t-1} \in \{1,0\}} p(V_t|V_{t-1}) p(V_{t-1}|S_{1:t-1}^N, Z_{1:t-1}) \tag{24}$$

VIII. Simulations and Results

We applied the tracking filter to a UAV video sequence of a vehicle moving in a cluttered urban environment. Some of the tracking parameters used are shown in Table I. The video data is a set of high fidelity simulations provided by General Dynamics. To validate the method, we manually identified the center of the vehicle as the true position of the target in the video sequence, frame-by-frame. Figure 7 shows the tracking results for the vehicle as it emerges from a full occlusion due to thick smoke. The drop in visibility can be seen in Panel (e) in

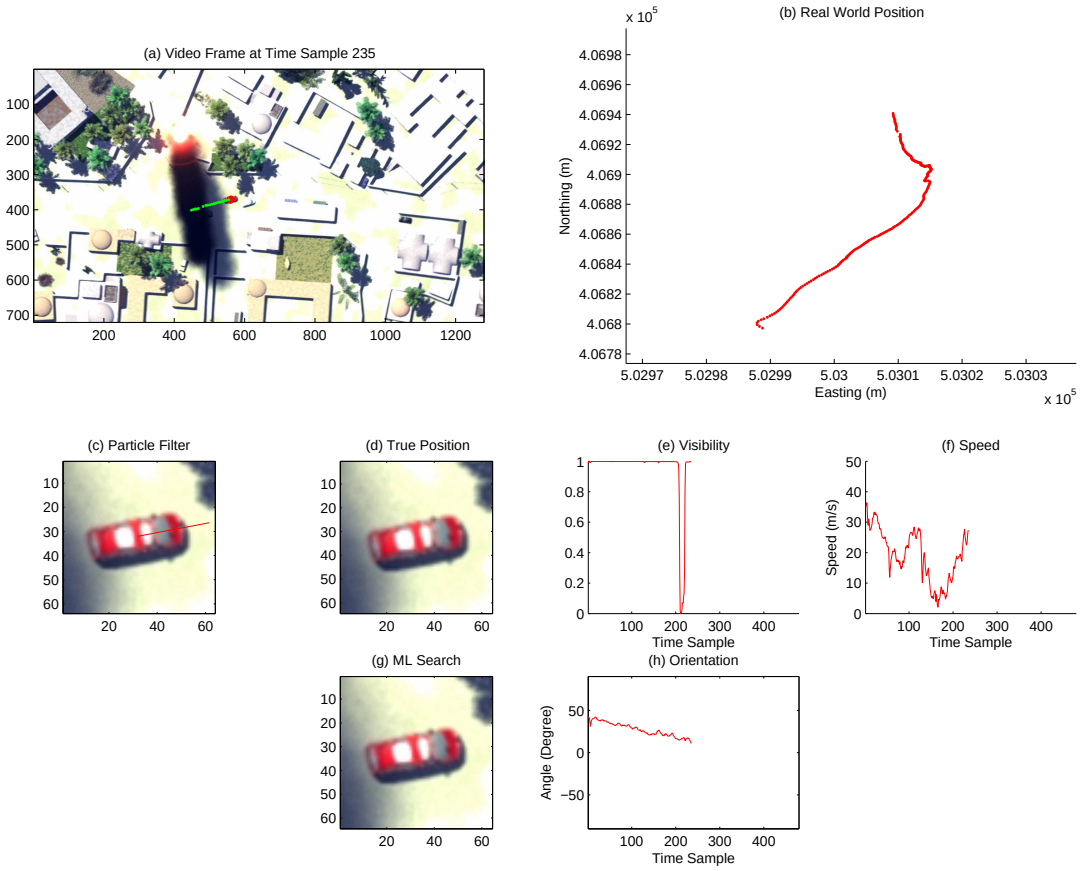


Fig. 7. Tracking results at Time Sample 235. This shows the target emerging from a full occlusion due to thick smoke. Panel (a) shows the estimated track (in green) and the posterior distribution of the particles (in red) for the position of the target in the video frame. Panel (b) shows a plot of the real world position of the target. Panel (c) shows the enlarged view of the target based on the current estimated position using the particle filter. Panel (d) shows the true position of the target, and Panel (g) shows the estimated position using the windowed ML search. Panel (e), (f) and (h) shows the estimated visibility, speed and orientation of the target respectively. Panel (e) shows clearly the decrease and subsequent increase in visibility as the target enter and emerge from the smoke occlusion.

Algorithm Parameter	**Symbol**	**Value**
Time interval between measurements	T	$\frac{1}{30}$ seconds
Number of Particles	N_p	200
Motion variance	q_x, q_y	625
Probability of becoming occluded	P_{NV}	0.1
Probability of becoming visible	P_V	0.1

TABLE I
TRACKING PARAMETERS FOR PARTICLE FILTER

Figure 7. Panel (d) shows the true position of the target in the image plane. Figure 8 shows the distribution of the particles as t he vehicle enters and emerges from the smoke occlusion. It can be seen that the posterior distribution of the vehicle's position increases significantly as it becomes occluded from view.

We have also compared the method to a windowed Maximum Likelihood (ML) search of the template in the video sequence. The windowed ML search method proceeds by forming a window, or neighbourhood, centered about the last known position of the target. For each time frame t, correlation scores Polar$(X_{Image}, Y_{Image}, \theta)$ are then computed at every point in this window. Finally, the maximum score with respect to $(X_{Image}, Y_{Image}, \theta)$ is taken as the new target location and orientation. If the score falls below a certain threshold value, the next window will be doubled in size and centered about the extrapolation of the previous two windows. This heuristic tool allows tracking to continue even when the target becomes temporarily, partially or fully occluded.

The result of the windowed ML search can also be seen in Panel (g) in Figure 7. Figure 9 compares the Root Mean Square Error (RMSE) of the windowed ML search and particle filter tracking with respect to the true position of the vehicle in the video sequence. They show similar performance. While the windowed ML search is similar to the result of the particle filter, it is very sensitive to the threshold value. If it is set too high, then legitimate targets may be ignored. The longer this happens, the more out of date the extrapolation becomes. On the other hand, if the threshold is set too low, then, when the

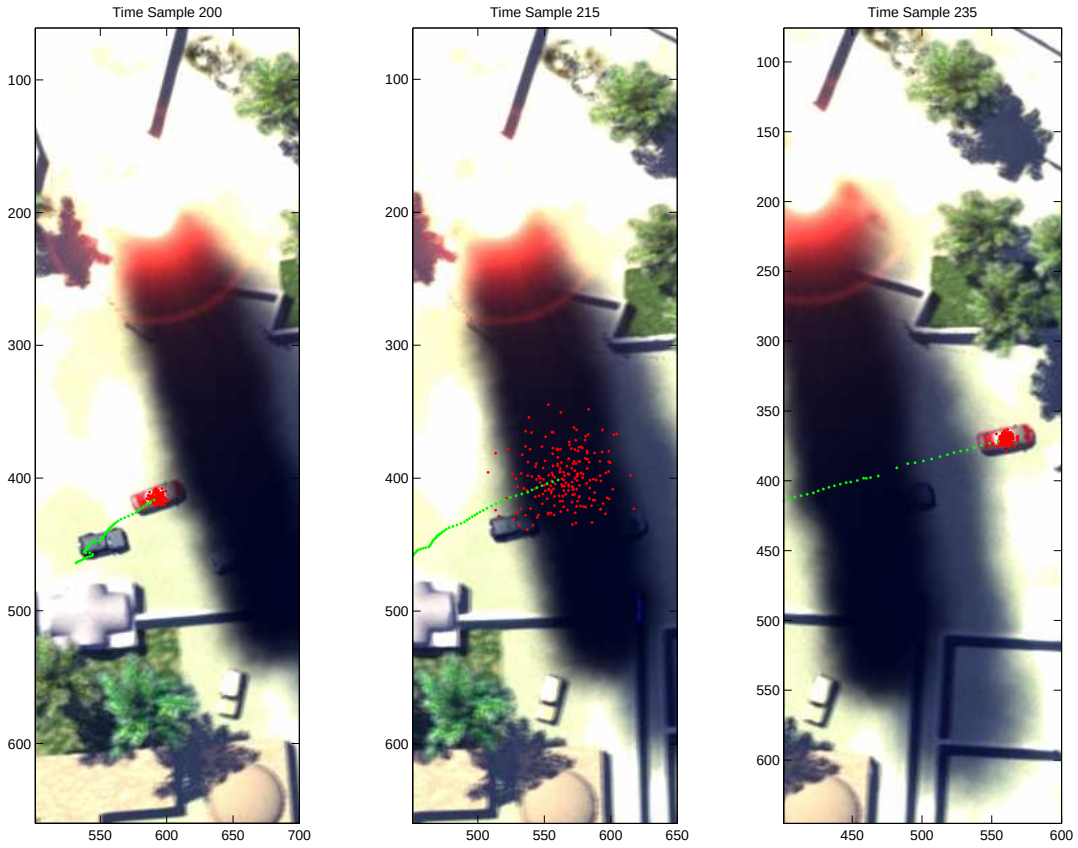


Fig. 8. A sequential series of frames showing the distribution of particles (in red) and estimated track (in green) as the target enters and emerges from the thick smoke occlusion. The posterior distribution of the vehicle's position increases significantly as it enters the occlusion.

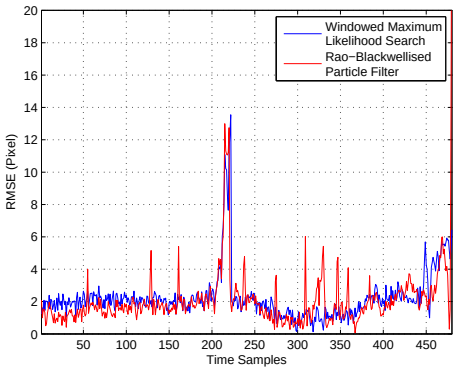


Fig. 9. This shows the Root Mean Square Error (RMSE) of the target's position in the video frame using the windowed ML search and particle filter method respectively

target becomes occluded, other nearby objects will register a higher score, and the algorithm will begin to follow false positives.

The particle filter tracker is more robust to local modes of the correlation surfaces of the polar matching algorithm. The particle distribution will expand to accommodate any uncertainty present. This explains the small spikes in the RMSE in Figure 9 as the posterior distribution stretches to account for uncertainty in the position of the target. A more suitable measure in this case with multi-modal posterior distribution might then be to consider the Maximum A Posteriori (MAP) estimate rather than the mean. As Figure 11 illustrates, both the maximum likelihood and particle filter methods achieve similar frame rates. However, the particle filter can successfully track the target in our example data with as few as 200 particles whereas the maximum likelihood method requires more than 400 points to work without permanently loosing the target. The conclusion drawn here is that the particle cloud can change in size and density, according to the probabilistic model, to cover a more optimal search area than the rigid, regularly sampled, square area utilised by the maximum likelihood approach.

IX. Conclusion

In this paper, we have shown that the combination of the rotation invariant dual-tree complex wavelet polar matching descriptor and the particle filter can be an effective approach to detect and track ground based targets from UAV sensor data. Polar matching offers target detection correlation scores for

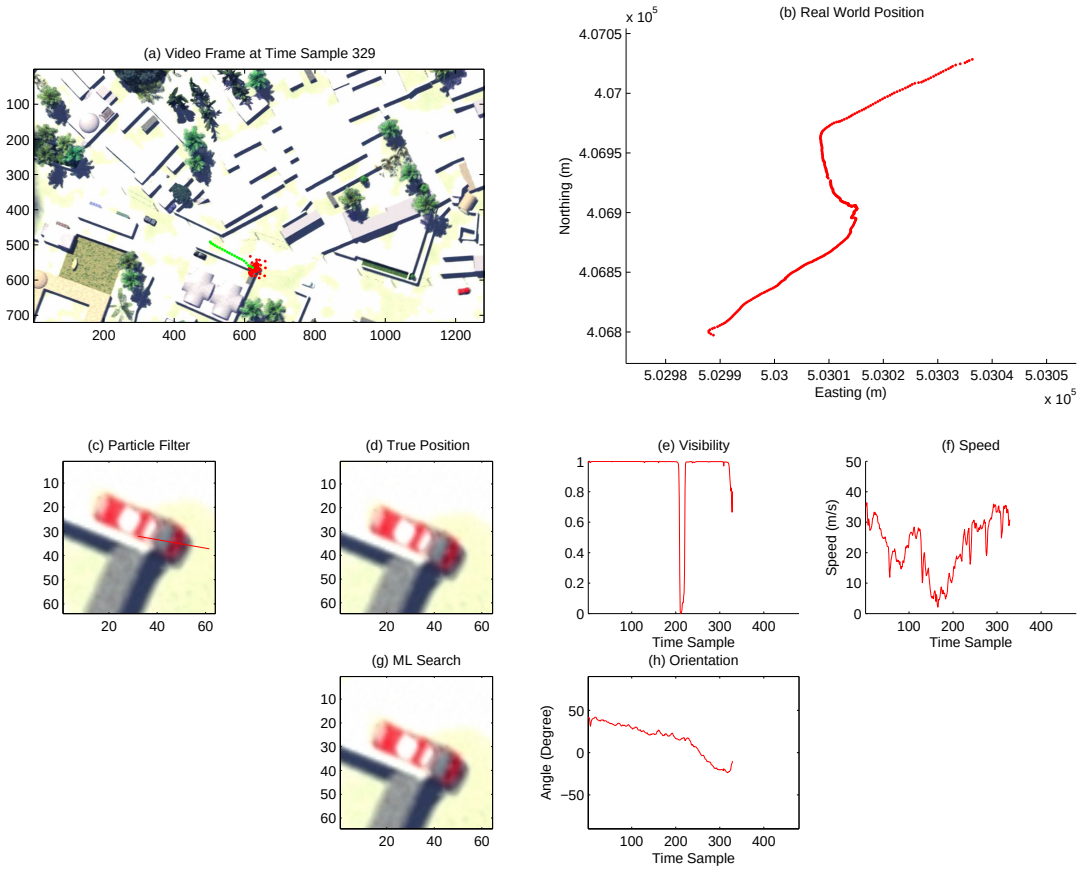


Fig. 10. Tracking results at Time Sample 329. This shows the target just as it is about to emerge from a partial occlusion due to a wall along the road. Panel (a) shows the estimated track (in green) and the posterior distribution of the particles (in red) for the position of the target in the video frame. Panel (b) shows a plot of the real world position of the target. Panel (c) shows the enlarged view of the target based on the current estimated position using the particle filter. Panel (d) shows the true position of the target, and Panel (g) shows the estimated position using the windowed ML search. Panel (e), (f) and (h) shows the estimated visibility, speed and orientation of the target respectively. Panel (a) shows that the particle cloud expanding around the target and the wall, which indicates some multi-modality behaviour of the posterior distribution of the estimated position of the targets.

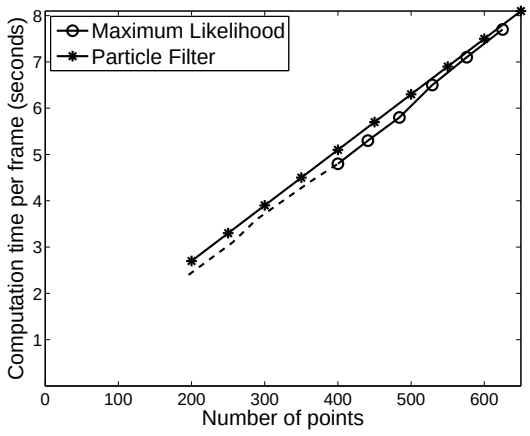


Fig. 11. The maximum likelihood and particle filter method require similar computational time. However, the particle filter can work with as few as 200 particles whereas the maximum likelihood method requires more than 400 points. The dotted line denotes tracking failure. (The times were calculated by running the respective Matlab implementations on a dual core 2.4 GHz Windows machine and averaging over 10 frames.)

each position and orientation to the particle filter. We studied the observation statistics using the polar matching and used a Track-Before-Detect likelihood to model the observation. This is further combined with a vehicle dynamical model and used in a Rao-Blakwellised Particle Filter tracking algorithm to track a vehicle in a video sequences. The particle filter provides good tracking of the target, and enhances the robustness of the tracking process. In the future, we will consider the case where the sensor's position and orientation are not known accurately and have to be estimated jointly with the target's position.

APPENDIX A
DERIVATION OF RAO-BLACKWELLISED PARTICLE FILTER (RBPF)

We use the $S_t^N = \{X_t, \theta_t\}$ for the general state and $S_t^L = \{V_t\}$ for the visibility variable which we marginalise in a RBPF.

In order to derive the RBPF, we first start with the joint

states $\{S^N_{1:t}, V_t\}$. Then, the joint distribution given the observation $Z_{1:t}$ can be written as

$$\begin{aligned}
&p(S^N_{1:t}, V_t|Z_{1:t}) \\
&= p(S^N_t, S^N_{1:t-1}, V_t|Z_t, Z_{1:t-1}) \\
&= p(Z_t|S^N_t, V_t, S^N_{1:t-1}, Z_{1:t-1}) p(S^N_t|S^N_{1:t-1}, V_t, Z_{1:t-1}) \\
&\times p(V_t|S^N_{1:t-1}, Z_{1:t-1}) p(S^N_{1:t-1}|Z_{1:t-1}) \frac{1}{p(Z_t|Z_{1:t-1})} \\
&= p(Z_t|S^N_t, V_t) p(S^N_t|S^N_{t-1}) p(V_t|S^N_{1:t-1}, Z_{1:t-1}) \\
&\times p(S^N_{1:t-1}|Z_{1:t-1}) \frac{1}{p(Z_t|Z_{1:t-1})}
\end{aligned} \tag{25}$$

To Rao-Blackwellised the above distribution, we consider the marginal $p(S^N_{1:t}|Z_{1:t})$,

$$\begin{aligned}
&p(S^N_{1:t}|Z_{1:t}) \\
&= \int p(S^N_{1:t}, V_t|Z_{1:t}) \mathrm{d}V_t \\
&= \left[\sum_{V_t \in \{1,0\}} p(Z_t|S^N_t, V_t) p(V_t|S^N_{1:t-1}, Z_{1:t-1}) \right] \\
&\times \frac{p(S^N_t|S^N_{t-1}) p(S^N_{1:t-1}|Z_{1:t-1})}{p(Z_t|Z_{1:t-1})} \\
&= p(Z_t|Z_{1:t-1}, S^N_{1:t}) \frac{p(S^N_t|S^N_{t-1}) p(S^N_{1:t-1}|Z_{1:t-1})}{p(Z_t|Z_{1:t-1})}
\end{aligned} \tag{26}$$

The above equation gives the weight update for a particle filter, assuming we have the sufficient statistic $p(V_{t-1}|S^N_{1:t-1}, Z_{1:t-1})$. Now we need to develop a probability update from time $t-1$ to time t.

Lets consider

$$\begin{aligned}
&p(V_t|S^N_{1:t}, Z_{1:t}) \\
&= \frac{p(V_t, Z_t, Z_{1:t-1}|S^N_{1:t})}{p(Z_{1:t}|S^N_{1:t})} \\
&= \frac{p(Z_t|S^N_{1:t}, V_t, Z_{1:t-1}) p(V_t|S^N_{1:t}, Z_{1:t-1}) p(Z_{1:t-1}|S^N_{1:t})}{p(Z_{1:t}|S^N_{1:t})} \\
&= \frac{p(Z_t|S^N_t, V_t) p(V_t|S^N_{1:t}, Z_{1:t-1})}{p(Z_t|Z_{1:t-1}, S^N_{1:t})}
\end{aligned} \tag{27}$$

And,

$$\begin{aligned}
&p(V_t|S^N_{1:t}, Z_{1:t-1}) \\
&= \sum_{V_{t-1} \in \{1,0\}} p(V_t|V_{t-1}) p(V_{t-1}|S^N_{1:t}, Z_{1:t-1}) \\
&= \sum_{V_{t-1} \in \{1,0\}} p(V_t|V_{t-1}) p(V_{t-1}|S^N_{1:t-1}, Z_{1:t-1})
\end{aligned} \tag{28}$$

as V_{t-1} is independent of S_t.

Also, we can calculate $p(Z_t|Z_{1:t-1}, S^N_{1:t})$ using the following equation:

$$\begin{aligned}
&p(Z_t|Z_{1:t-1}, S^N_{1:t}) \\
&= \sum_{V_t \in \{1,0\}} p(Z_t|S^N_{1:t}, V_t) p(V_t|S^N_{1:t}, Z_{1:t-1}) \\
&= \sum_{V_t \in \{1,0\}} p(Z_t|S^N_t, V_t) p(V_t|S^N_{1:t}, Z_{1:t-1})
\end{aligned} \tag{29}$$

From Eq. 27, 28 and 29, we can update the sufficient statistic $p(V_t|Z_{1:t}, S^N_{1:t})$.

We can estimate the $p(V_t|Z_{1:t})$ using the particle approximation of $p(S^N_{1:t}|Z_{1:t})$

$$\begin{aligned}
p(V_t|Z_{1:t}) &= \int p(V_t|Z_{1:t}, S^N_{1:t}) p(S^N_{1:t}|Z_{1:t}) \mathrm{d}S^N_{1:t} \\
&\approx \sum_{p=1}^{N_p} p(V_t|S^N_{1:t}, Z_{1:t}) w_{t,p} \ \delta(S^N_{1:t} - S^N_{1:t,p}) \\
&= \sum_{p=1}^{N} w_{t,p} \ p(V_t|S^N_{1:t,p}, Z_{1:t})
\end{aligned} \tag{30}$$

ACKNOWLEDGMENT

This research was supported by the Data and Information Fusion Defence Technology Centre, UK, under the Fusion and Tracking Clusters. The authors thank these parties for funding this work and would like to thank General Dynamics, UK for providing the synthetic data.

REFERENCES

[1] N. G. Kingsbury, "Rotation-Invariant Local Feature Matching with Complex Wavelets," *Proc. European Conference on Signal Processing (EUSIPCO)*, September 2006.

[2] Y. Bar-Shalom and W. D. Blair, Eds., *Multitarget-Multisensor Tracking: Applications and Advances*. 685 Canton Street, Norwood, MA 02062: Artech House, 2000, vol. III.

[3] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. 685 Canton Street, Norwood, MA 02062: Artech House, 1999.

[4] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, pp. 107–113, April 1993.

[5] A. Doucet, S. Godsill, and C. Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, vol. 10, pp. 197–208, 2000.

[6] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods - Second Edition*. New York: Springer, 2004.

[7] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice*. Chapman and Hall/CRC, 1996.

[8] N. G. Kingsbury, "Complex Wavelets for Shift Invariant Analysis and Filtering of Signals," *Journal of Applied and Computational Harmonic Analysis*, vol. 10, pp. 234–253, May 2001.

[9] D. G. Lowe, "Distinctive Image Features from Scale-invariant Keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[10] E. P. Simoncelli and W. T. Freeman, "The Steerable Pyramid: A Flexible Architecture for Multi-scale Derivative Computation," *Proc. ICIP*, vol. 3, pp. 444–447, October 1995.

[11] X. R. Li and V. P. Jilkov, "Survey of Maneuvering Target Tracking. Part I: Dynamic Models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, p. 13331364, 2003.

[12] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter - Particle Filters for Tracking Applications*. 685 Canton Street, Norwood, MA 02062: Artech House, 2004.

[13] S. K. Pang, J. Li, and S. Godsill, "Models and Algorithms for Detection and Tracking of Coordinated Groups," *IEE Aerospace Conference*, March 2008.

[14] A. Kong, J. Liu, and W. Wong., "Sequential Imputation and Bayesian Missing Data Problems," *J. Americian Statistical Association*, pp. 278–288, 1994.
[15] G. Kitagawa, "Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models," *Journal of Computational and Graphical Statistics*, vol. 5, pp. 1–25, 1996.
[16] J. S. Liu and R. Chen, "Sequential Monte Carlo Methods for Dynamic Systems," *J. Americian Statistical Association*, vol. 93, p. 10321044, 1998.
[17] O. Cappe, S. Godsill, and E. Moulines, "An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo," *Proceedings of the IEEE*, vol. 95, May 2007.

Sze Kim, Pang received the M.Eng. degree in electrical and electronic engineeing from Imperial College, UK in 1997. Since 1999, he has been with DSO National Laboratories, Singapore, where he is now a Senior Member of Technical Staff. His work covers research and development in statistical and array signal processing. Currently, he is pursuing a Ph.D. Degree in the Signal Processing and Communications Laboratory, Cambridge University Engineering Department. His research focuses on multitarget detection and tracking using Bayesian statistical signal processing.

James D. B. Nelson is a research associate in the Signal Processing and Communications Laboratory at Cambridge University. Prior to this he held post-doctoral research positions at Cranfield University (3 years) and Southampton University (2 years). His research interests include: wavelets and multiresolution analysis, information fusion, sampling theory and reconstruction, and support vector machines.

Simon J. Godsill is Professor in Statistical Signal Processing in the Engineering Department of Cambridge University. He is an Associate Editor for IEEE Trans. Signal Processing and the journal Bayesian Analysis, and is a member of IEEE Signal Processing Theory and Methods Committee. He has research interests in Bayesian and statistical methods for signal processing, Monte Carlo algorithms for Bayesian problems, modelling and enhancement of audio and musical signals, source separation, tracking and genomic signal processing. He has published extensively in journals, books and conferences. He has co-edited in 2002 a special issue of IEEE Trans. Signal Processing on Monte Carlo Methods in Signal Processing and a recent special issue of the Journal of Applied Signal Processing, and organised many conference sessions on related themes.

PLACE
PHOTO
HERE

Nick G. Kingsbury