

# VIDEO TRACKING USING DUAL-TREE WAVELET POLAR MATCHING AND PARTICLE FILTERING

S. K. Pang, J. D. B. Nelson, S. J. Godsill, and N. G. Kingsbury  
University of Cambridge  
Signal Processing and Communications Laboratory  
CUED, Trumpington Street, Cambridge CB2 1PZ  
skp31, jdbn2, sjg, ngk @eng.cam.ac.uk

**Keywords:** Dual-Tree Wavelet, Polar Matching, Video Tracking, Particle Filtering

## Abstract

In this paper, we describe a video tracking application using the dual-tree polar matching algorithm. The models are specified in a probabilistic setting, and a particle filter is used to perform 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.

## 1. Introduction

Detection and tracking of a known target in video sequences is a common and important problem in image processing. In this paper, we will 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 brief period of 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 scale variations.

With this in mind, the descriptor and matching technique afforded by rotation-invariant polar matching with dual-tree complex wavelet transforms (DTCWT) recently developed by Kingsbury in 2006 [6] is adapted here, for the first time, 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 [1][2]. With the parallel advances in modern computational power and the developments in optimal non-linear techniques such as particle filters [5][3] and Markov Chain Monte Carlo (MCMC) [15][4], it is now possible to consider the exploitation of other information (such as non-linear measurement process) 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 have designed a particle filter to perform the tracking.

The paper is organised as follows. Section 2 presents rotation-invariant dual-tree complex wavelet polar matching. Section 3 describe the probabilistic state-space model. Section 4 and 5 describe the dynamic models and observation model respectively. Section 6 describes the particle filter algorithm. Simulation results are shown in Section 7, followed by conclusions in Section 8.

## 2. Polar matching

Extending his work on the shift-invariant dual-tree complex wavelet transform [7], Kingsbury recently introduced the rotation-invariant polar matching method [6]. Owing to low redundancy, the DTCWT descriptor is more efficient than the existing popular scale- and rotation-invariant methods of SIFT [12] and Simoncelli's steerable pyramids [17]. 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.

Kingsbury's method proceeds by firstly computing the DTCWT coefficients of a template. The centre of the target

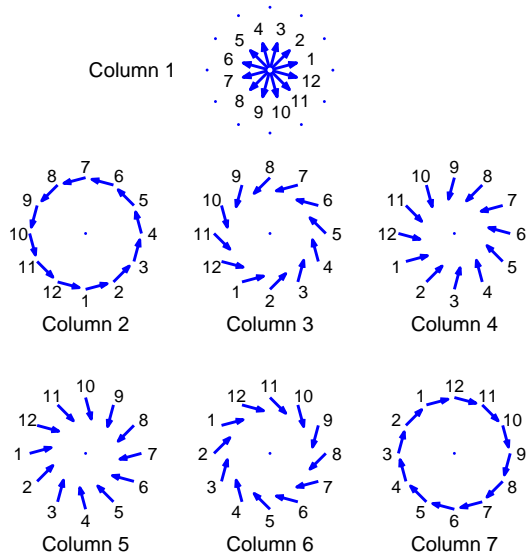


Figure 1. Locations and orientations of the DTCWT coefficients. Each orientation describes a coefficient, or conjugate, of one of the six subbands. The column numbers indicate the column of the P-matrix in which the coefficients are placed, and the numbers displayed around each circle indicate the row of the P-matrix that each coefficient is placed. Taken from [6].

is located manually and the complex wavelet coefficients at this point are stored. Coefficients are also taken around one or more circles, about the centre point, at 30 degree increments and at multiple scales. As Figure 1 illustrates, the coefficients are then arranged into a polar matching matrix (P-matrix) such that a rotation of  $k \times 30^\circ$  in the original image will manifest a vertical shift by  $k$  rows in the P-matrix. Consider two images, one a  $30n^\circ$  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  $n$ .

However, the rotational sensitivity can be increased to  $7.5^\circ$  via careful band-limited interpolation. This is achieved by performing the correlation as a product in the Fourier domain and zero padding. Care should be taken here. The first column of a P-matrix, formed about the centre of a single step edge will vary slowly as the edge is rotated. Columns 2 and 7 will vary quicker, 3 and 6 quicker still and 4 and 5 quickest of all. Hence, the zeros must be placed according to P-matrix column. 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 rotation invariance from the P-matrix construction.

The Polar Matching Algorithm will provide a correlation score for a template in a specific position  $(X_{Image}, Y_{Image})$

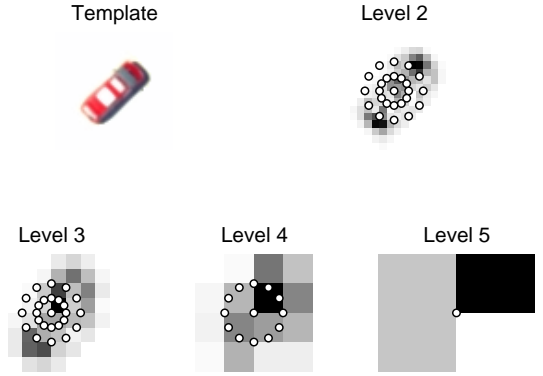


Figure 2. DTCWT coefficients of a template. The white dots indicate the locations of the coefficients that are used in the P-matrix template. Four different scales, or levels, are used. Only one of one of the 6 subbands is shown.

and orientation  $\theta$  within a larger video frame. The polar matching function is referred to as  $\text{Polar}(X_{Image}, Y_{Image}, \theta)$ . For the video tracking application, the template will be provided as an image taken of the vehicle of interest.

### 3. 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,  $\theta$ , 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  $X_t = [x_t \dot{x}_t y_t \dot{y}_t \theta_t V_t]$ .

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

$$p(X_t|Z_{1:t}) = \frac{p(z_t|X_t)p(X_t|Z_{1:t-1})}{p(z_t|Z_{1:t-1})} \quad (1)$$

$$p(X_t|Z_{1:t-1}) = \int p(X_t|X_{t-1})p(X_{t-1}|Z_{1:t-1})dX_{t-1} \quad (2)$$

where  $Z_{1:t} = [z_1 \dots z_m \dots z_t]$  and  $z_m$  denotes all of the observations collected at time  $m$ .

### 4. Dynamical Models

We choose to write the transition probability model  $p(X_t|X_{t-1})$  as

$$p(X_t|X_{t-1}) = p(S_t|S_{t-1})p(\theta_t|\theta_{t-1})p(V_t|V_{t-1}) \quad (3)$$

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

For the target dynamic, we will use the discrete time equivalent of the near constant velocity model [10]. This is given by

$$S_t = FS_{t-1} + w_t \quad (4)$$

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

where  $w_t$  is a Gaussian noise with covariance  $Q_w$  given by

$$Q_w = \begin{bmatrix} \frac{T^3}{3}S_x & \frac{T^2}{2}S_x & 0 & 0 \\ \frac{T^2}{2}S_x & TS_x & 0 & 0 \\ 0 & 0 & \frac{T^3}{3}S_y & \frac{T^2}{2}S_y \\ 0 & 0 & \frac{T^2}{2}S_y & TS_y \end{bmatrix} \quad (6)$$

with  $S_x$  and  $S_y$  being the variance of the driving noise of the dynamical process. Hence,

$$S_t|S_{t-1} \sim N(FS_{t-1}, Q_w) \quad (7)$$

The orientation  $\theta_t$  models the changes in the orientation of the target at time  $t$ . This provides us with the flexibility to track changes in the image plane as the vehicle rotates in plane. This is modeled as a random walk:

$$\theta_t|\theta_{t-1} \sim N(\theta_{t-1}, S_\theta) \quad (8)$$

where  $S_\theta$  is the variance of the driving noise. The visibility variable  $V_t$  tries to determine if the target is visible, or if it is temporarily obscured by smoke or walls. This affects the way the likelihood term  $p(z_t|X_t)$  is estimated. The target's visibility variable will be modeled as a discrete Markov chain,

$$p(V_t = 0|V_{t-1} = 1) = P_{NV} \quad (9)$$

$$p(V_t = 1|V_{t-1} = 0) = P_V \quad (10)$$

## 5. Observation Model

At each time step, the observation  $z_t$  describes one video frame. A principled approach to calculating the likelihood probability  $p(z_t|X_t)$  will need to consider the probability distribution of the target and the background clutter with respect to the polar matching function and chosen template. These can then be used in a likelihood ratio form that is common in track-before-detect applications [14][16]. Here we approximate the process using the following likelihood ratio function:

$$p(z_t|X_t) \propto \begin{cases} \exp(k \times \text{Polar}(H(x_t, y_t), \theta_t)) & \text{if } V_t = 1; \\ \exp(V_c) & \text{Otherwise.} \end{cases} \quad (11)$$

In this paper, we assume that the position and the orientation of the sensor to be known. The non-linear function  $(X_{Image}, Y_{Image}) = H(X, Y)$  maps the grid coordinates of the target to the image plane of the sensor. The polar matching function  $\text{Polar}(\cdot)$ , as described in Section 2, returns the correlation score with respect to the position  $(X_{Image}, Y_{Image})$  and orientation  $\theta$ . The constant  $k$  is a scaling factor. The visibility variable  $V_c$  is chosen such that it is, on average, higher than the background correlation score, but much less than the self correlation score of the image template. This gives the tracking algorithm the ability to switch to an occluded state.

The exponential form of the likelihood term is used because it gives more emphasis on the larger values of the correlation function  $\text{Polar}(\cdot)$ . Other forms have also been experimented with, including flooring the negative correlation score to zero. In [13], more discussions on the optimal form of linear likelihood functions in the presence of noise are discussed. However, in the context of the polar matching algorithm, this is the subject of further work.

## 6. 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 [14] 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  $\{X_{t,p}, w_{t,p}\}_{p=1}^N$ . 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

$$p(X_t|Z_{1:t}) \approx \sum_{p=1}^N w_{t,p} \delta(X_t - X_{t,p}) \quad (12)$$

where  $Z_{1:t} = [z_1 \dots z_m \dots z_t]$  are the observations and the weight  $w_{t,p}$ , of the particle  $p$ , is updated according to

$$w_{t,p} = w_{t-1,p} \times \frac{p(z_t|X_{t,p})p(X_{t,p}|X_{t-1,p})}{q(X_{t,p}|X_{t-1,p}, z_t)} \quad (13)$$

The choice of the importance density  $q(X_{t,p}|X_{t-1,p}, z_t)$  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  $X_{t-1,p}$  and  $z_t$  is  $p(X_{t,p}|X_{t-1,p}, z_t)$  [3]. There are other suboptimal choices. For example, a popular choice is to use the prior model density  $p(X_{t,p}|X_{t-1,p})$ . When substituted into Equation (13), we obtain

$$w_{t,p} = w_{t-1,p} \times p(z_t|X_{t,p}) \quad (14)$$

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 [3]. This will adversely affect the accuracy and lead to the degeneracy problem where, after a certain number 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  $p(X_t|Z_{1:t})$  is almost zero. A practical measure of the degeneracy of the particle weights is the effective sample size  $N_{eff}$  introduced in [9]:

$$\hat{N}_{eff} = \left( \sum_{p=1}^N w_{t,p}^2 \right)^{-1} \quad (15)$$

It is easy to see that  $1 \leq N_{eff} \leq N$ . 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 [5] has to be performed. Resampling eliminates sample with low weights and multiplies samples with high importance weights. This involves mapping a random measure  $\{X_{t,p}, w_{t,p}\}_{p=1}^N$  into a random measure  $\{X_{t,p}, \frac{1}{N}\}_{p=1}^N$  with uniform weights.

There are several methods available when implementing the remapping step. The first introduction of resampling [5] 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 [8] and residual sampling [11] may be applied. Systematic Resampling [8] is another efficient method. It is simple to implement, it has order  $N$  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  $p(X_t|X_{t-1})$  as the importance function. For the resampling step, we use Systematic Resampling.

## 7. Simulations and Results

We applied the tracking filter to a UAV video sequence, tracking a vehicle, moving in a cluttered urban environment. The video data is a set of high fidelity simulations provided by General Dynamics. The true position of the target in the video sequence is identified manually frame-by-frame as the center of the vehicle. Some of these tracking parameters are shown in Table 1. Figure 3 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 Figure 3. Figure 4 shows the distribution of the particles as the vehicle enters and emerges from the smoke occlusion. The posterior distribution of the vehicle's position increases significantly as it becomes occluded from view. Panel (d) shows the true position of the target in the image plane. We have also compared the method

Algorithm Parameter	Symbol	Value
Time interval between measurements	$T$	$\frac{1}{30}$ seconds
Number of Particles		300
Motion variance	$S_x, S_y$	1600
Likelihood Scale	$k$	5
Probability of becoming occluded	$P_{NV}$	0.1
Probability of becoming visible	$P_V$	0.25
Likelihood Constant (when occluded)	$V_C$	0.2

Table 1. Tracking Parameters for Particle Filter

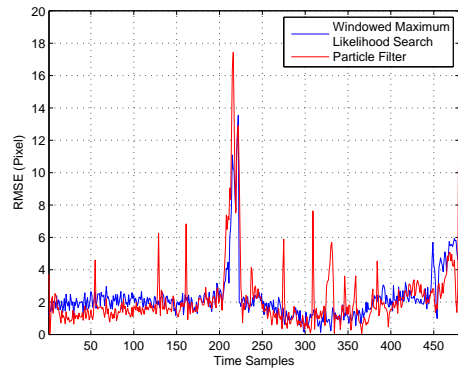


Figure 5. This shows the Root Mean Square Error (RMSE) of the target's position in the video frame using the windowed ML search and particle filter method respectively

with 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  $\text{Polar}(x, y, \theta)$  are then computed at every point in this window. Finally, the maximum score with respect to  $(x, y)$  is taken as the new target location, and the maximum with respect to  $\theta$  is the target orientation, relative to the original template. 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 the target to become, temporarily, partially or fully occluded.

The result of the windowed ML search can also be seen in Panel (g) in Figure 3. Figure 5 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 sequences. 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. The problem here is that the choice of the threshold

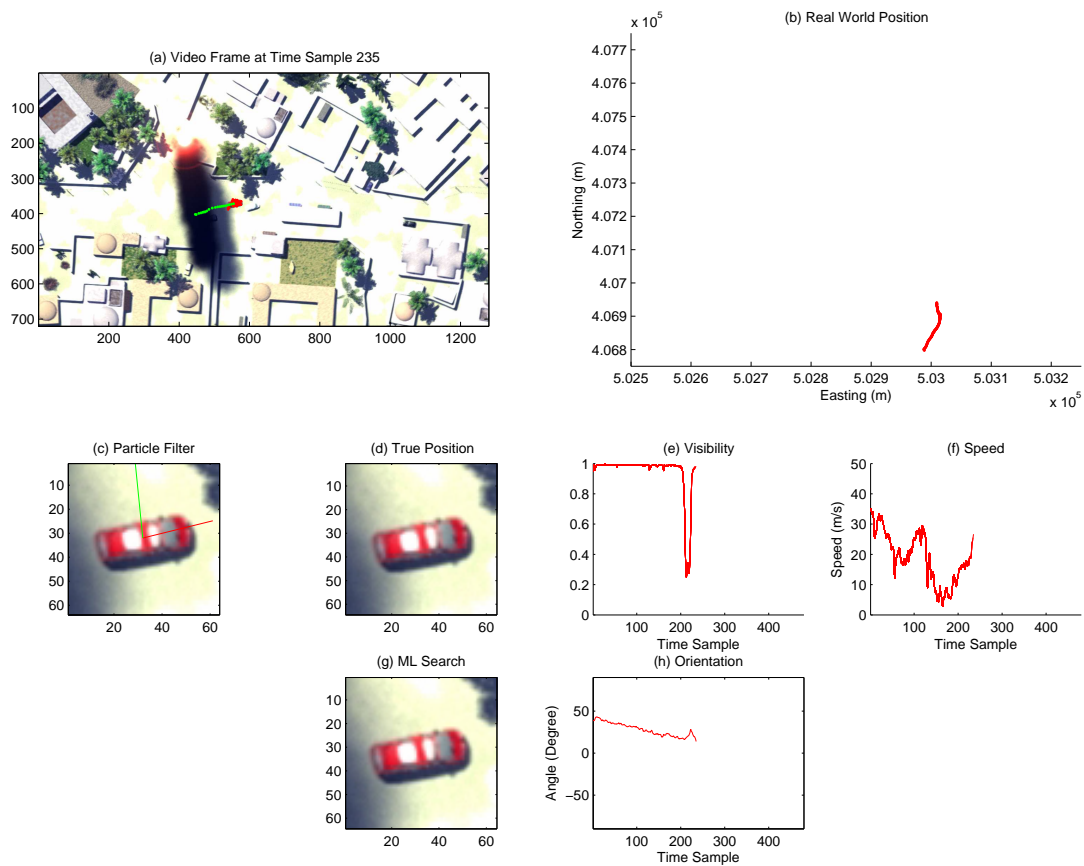


Figure 3. Tracking results at Time Sample 235. This shows the target emerging from a full occlusion due to thick smoke. Panel (a) shows the estimated track (in green) and the posterior distribution of the particles (in red) for the position of the target in the video frame. Panel (b) shows a plot of 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.

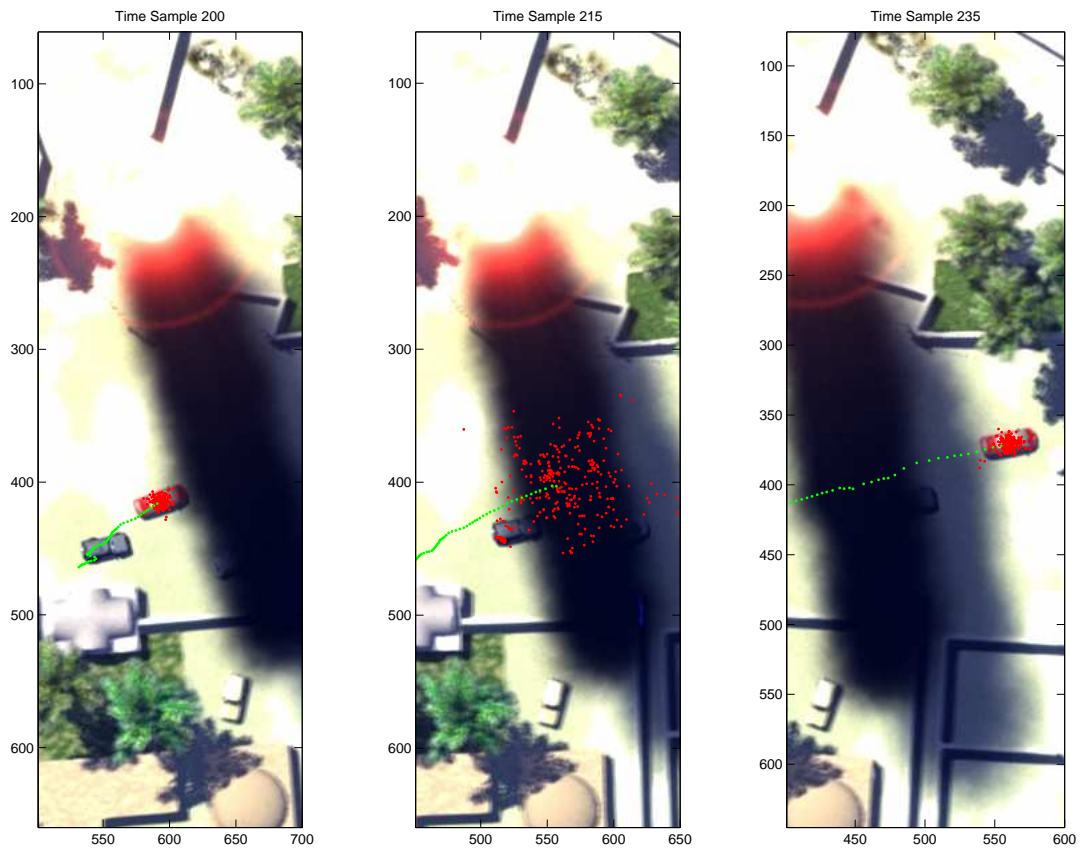


Figure 4. A sequential series of frames showing the distribution of particles (in red) and estimated track (in green) as the target enters and emerges from the thick smoke occlusion. The posterior distribution of the vehicle's position increases significantly as it enters the occlusion.

can be critical. 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 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. Figure 6 shows the target as it is partially occluded by a wall. The particle cloud can be seen to have split into two, highlighting the presence of two possible modes in the posterior distribution of the position of the target. The particle cloud subsequently converges back to the moving target after a few video frames. This is because the video data shows that the moving target is a more likely scenario, instead of the target being hidden under a wall.

This explains the small spikes in the RMSE in Figure 5 as the posterior distribution stretches or split into two modes to account for uncertainty in the position of the target. A more suitable measure in this case with multi-modal posterior distribution will be to consider the maximum a posteriori (MAP) estimate rather than the mean.

## 8. Conclusions and Future Works

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 each position and orientation to the particle filter. This is used in a 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 has to be estimated jointly with the target's position.

## 9. Acknowledgments

This research was sponsored by the Data and Information Fusion Defence Technology Centre, UK, under the Tracking Cluster. The authors thank these parties for funding this work. The authors will also like thank General Dynamics for providing the high fidelity synthetic data. The authors are grateful to Simon Maskell for the discussions on the form of likelihood function.

## References

- [1] Y. Bar-Shalom and W. D. Blair, editors, *Multitarget-Multisensor Tracking: Applications and Advances*, volume

- III, Artech House, 685 Canton Street, Norwood, MA 02062, 2000.
- [2] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, 685 Canton Street, Norwood, MA 02062, 1999.
- [3] A. Doucet, S. Godsill, and C. Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering", *Statistics and Computing*, 10, 2000, pp. 197–208.
- [4] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice*, Chapman and Hall/CRC, 1996.
- [5] 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*, 140, April 1993, pp. 107–113.
- [6] N. G. Kingsbury, "Rotation-Invariant Local Feature Matching with Complex Wavelets", *Proc. European Conference on Signal Processing (EUSIPCO)*, September 2006.
- [7] N. G. Kingsbury, "Complex Wavelets for Shift Invariant Analysis and Filtering of Signals", *Journal of Applied and Computational Harmonic Analysis*, 10, May 2001, pp. 234–253.
- [8] G. Kitagawa, "Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models", *Journal of Computational and Graphical Statistics*, 5, 1996, pp. 1–25.
- [9] A. Kong, J. Liu, and W. Wong., "Sequential Imputation and Bayesian Missing Data Problems", *J. American Statistical Association*, 1994, pp. 278–288.
- [10] X. R. Li and V. P. Jilkov, "Survey of Maneuvering Target Tracking. Part I: Dynamic Models", *IEEE Transactions on Aerospace and Electronic Systems*, 39, 2003, pp. 13331364.
- [11] J. S. Liu and R. Chen, "Sequential Monte Carlo Methods for Dynamic Systems", *J. American Statistical Association*, 93, 1998, pp. 10321044.
- [12] D. G. Lowe, "Distinctive Image Features from Scale-invariant Keypoints", *International Journal of Computer Vision*, 60, 2004, pp. 13331364.
- [13] S. Maskell, "A Bayesian Approach to Fusing Uncertain, Imprecise and Conflicting Information", *Information, Fusion*, 2007.
- [14] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter - Particle Filters for Tracking Applications*, Artech House, 685 Canton Street, Norwood, MA 02062, 2004.
- [15] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods - Second Edition*, Springer, New York, 2004.
- [16] M. Rutten, N. Gordon, and S. Maskell, "Particle-Based Track-Before-Detect in Rayleigh Noise", *Proc. SPIE Conf. on Signal and Data Processing of Small Targets*, 2004.
- [17] E. P. Simoncelli and W. T. Freeman, "The Steerable Pyramid: A Flexible Architecture for Multi-scale Derivative Computation", *Proc. ICIP*, 3, October 1995, pp. 444–447.



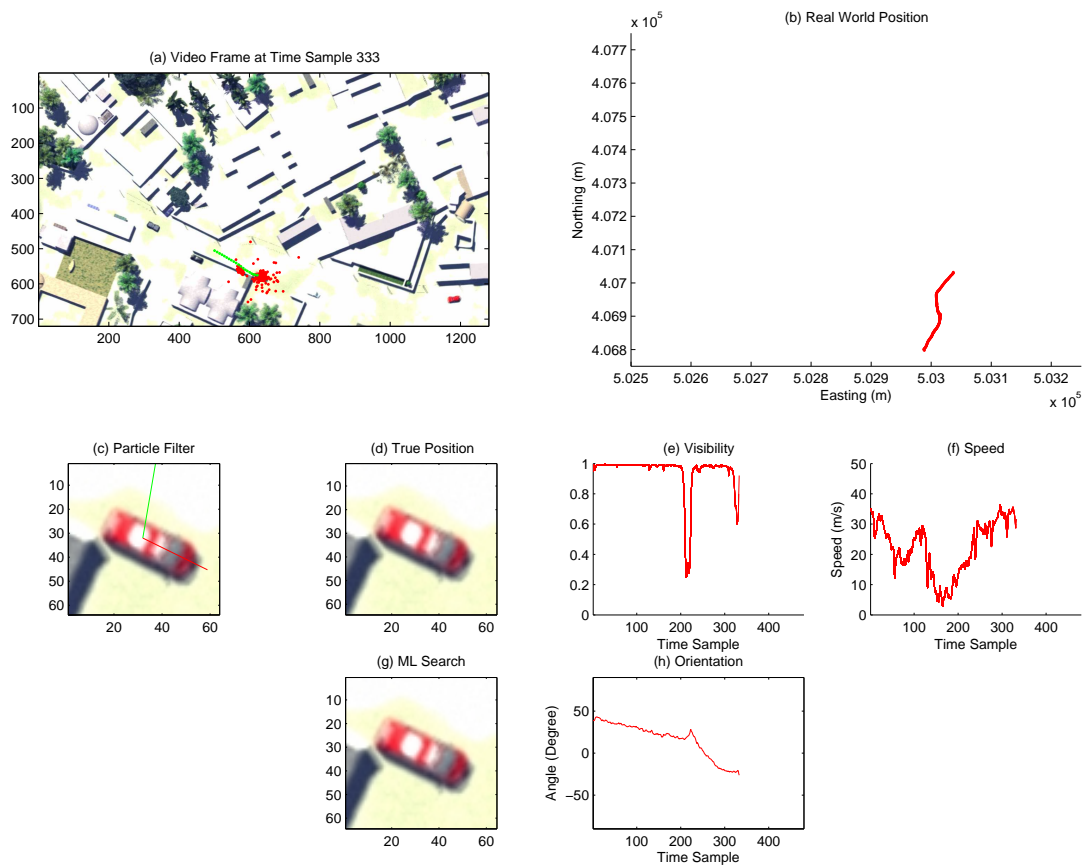


Figure 6. Tracking results at Time Sample 333. This shows the target just as it is about to emerge from a partial occlusion due to a wall along the road. Panel (a) shows the estimated track (in green) and the posterior distribution of the particles (in red) for the position of the target in the video 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 clearly the two modes in the posterior distribution of the estimated position of the targets.