Structural Similarity-Based Object Tracking in Video Sequences

Artur Łoza*, Lyudmila Mihaylova**, Nishan Canagarajah* and David Bull*

* Department of Electrical and Electronic Engineering, University of Bristol, UK
** Department of Communication Systems, Lancaster University, UK
[artur.loza, nishan.canagarajah, dave.bull]@bristol.ac.uk, mila.mihaylova@ieee.org

Abstract - ***This paper addresses the problem of object tracking in video sequences. The use of a structural similarity measure for tracking is proposed. The measure reflects the distance between two images by comparing their structural and spatial characteristics and has shown to be robust to illumination and contrast changes. As a result it guarantees robustness of the tracking process under changes in the environment. The previously used Bhattacharyya distance is not robust to such changes. Additionally, when a tracker is run with the Bhattacharyya distance, histograms should be calculated in order to find the likelihood function of the measurements. With the new function there is no need to calculate histograms. A particle filter (PF) is implemented where this measure is used for computing the distance between the reference and current frame. The algorithm performance has been tested and evaluated over real-world video sequences, and has been shown to outperform methods based on colour and edge histograms.***

Keywords: Similarity measure, object tracking, video sequences, particle filtering.

1 Introduction

Recently there has been an increasing interest in target tracking in video sequences. This problem faces many challenges, some of them are related to the models of the moving object, the measurement model and the function characterising the similarity between two images/video frames. One of the particularities of object tracking in video sequences compared to tracking with radar data is that there is no measurement model in explicit form. Some image features, such as colour, motion, and edges, can be used to track the moving object [8]. The performance of the tracking algorithm depends also on the measure characterising the similarity or dissimilarity between the two subsequent images/video frames. Often used functions are the Bhattacharyya distance [1, 4] and the non-metric Kullback-Leibler measure.

In this paper we propose the use of structural similarity measure for object tracking in video sequences by means of a particle filter. The motivation of applying particle filtering is that it has been proven to be a scalable and powerful approach, able to cope with nonlinearities, and work under uncertainties, which makes it a suitable approach for object tracking in video sequences (see for example [8] and [3]). The similarity measure proposed in [11] captures spatial characteristics of an image and has shown to be robust to illumination and contrast changes. It has been used for the purposes of quality assessment of distorted and fused images [6, 9], but not for tracking. In the present paper, we show how this measure can be applied for tracking purposes. It allows one to substitute histograms and to calculate in a straightforward way the measurement likelihood function within particle filtering. We show that it is a good and fast alternative to histogram based tracking.

The remaining part of the paper is organised as follows. Section 2 presents the image similarity measure for tracking. Section 3 describes the motion model of the region surrounding the object of interest and the likelihood model. Section 4 presents a particle filter with the proposed similarity measure. Section 5 contains results over real-world video sequences. Finally, Section 6 discusses the results and open issues for future research.

2 Distance measure

2.1 Structural similarity measure

The proposed method uses a similarity measure computed directly in the image spatial domain. This approach differs significantly from most of the particle filter algorithms, that compare image distributions represented by their sample histograms [8].

Although many simple image similarity measures exist, for example, Minimum Mean Square Error, Mean Absolute Error or Peak-Signal to Noise Ratio, most of these mathematical measures have failed to capture the perceptual similarity of images when subjected to varying luminance, contrast, compression or noise [11]. Recently, based on the premise that the human visual system is highly tuned to extracting structural information, a new image metric has been developed, called the Structural SIMilarity (SSIM) Index [11]. The SSIM index, S, between two images, $\mathbf{a}$

and $\mathbf{b}$ is defined as follows:

$$S(\mathbf{a},\mathbf{b}) = \left(\frac{2\mu_a\mu_b}{\mu_a^2+\mu_b^2}\right)^{\beta}\left(\frac{2\sigma_a\sigma_b}{\sigma_a^2+\sigma_b^2}\right)^{\alpha}\left(\frac{\sigma_{ab}}{\sigma_a\sigma_b}\right)^{\gamma}, \quad (1)$$

where μ and σ stand for mean and sample standard deviation, respectively, and σ_{ab} corresponds to sample covariance. The three components of S, reading from the left, measure how close the luminance, contrast and structural similarity of the two images are. Such a combination of the three image properties can be seen as a case of a image cue fusion. The exponents $\alpha,\beta,\gamma \geq 0,\ \alpha{+}\beta{+}\gamma > 0$ are used to adjust the impact of each measurement on the final value of S.

It can easily be shown that the measure defined in (1) is symmetric and has a unique upper bound: $S(\mathbf{a},\mathbf{b}) \leq c_0,\ S(\mathbf{a},\mathbf{b}) = c_0 = 1$ iff $\mathbf{a}=\mathbf{b}$. For detailed analysis of the SSIM measure, the reader is referred to [11].

2.2 Image dissimilarity

Below, we present a method of evaluating the likelihood function $\mathcal{L}$ (see Section 3), based on the similarity between two grayscale images, represented here as vectors formed from the image regions. One of the ways to convert similarity $S(\mathbf{a},\mathbf{b})$ into normalised dissimilarity $D(\mathbf{a},\mathbf{b})$ is as follows [12]:

$$D(\mathbf{a},\mathbf{b}) = \frac{c_0 - S(\mathbf{a},\mathbf{b})}{c_1},$$

where c_0 and c_1 are chosen to map a distance into the interval $[0,1]$. An alternative way [12],

$$D(\mathbf{a},\mathbf{b}) = \frac{c_0}{S(\mathbf{a},\mathbf{b})} - 1 \quad (2)$$

is preferred, however, as it only requires knowledge of maximal value of S and is more sensitive to very dissimilar vectors. The dissimilarity between images used in the method proposed in this paper is obtained by substituting (1) into (2) (as noted in the previous paragraph, $c_0 = 1$):

$$D(\mathbf{a},\mathbf{b}) = \left(\frac{2\mu_a\mu_b}{\mu_a^2+\mu_b^2}\right)^{-\alpha}\left(\frac{\sigma_{ab}}{\sigma_a\sigma_b}\right)^{-\beta}\left(\frac{2\sigma_a\sigma_b}{\sigma_a^2+\sigma_b^2}\right)^{-\gamma} - 1. \quad (3)$$

It can be shown that this measure satisfies nonnegativity (if the absolute value of sample covariance is used), reflexivity and symmetry conditions. For a dissimilarity measure to be a metric (distance) a triangle inequality has to be satisfied. However, for our purposes, the descriptiveness and discriminating ability of the measure are sufficient and this condition is not verified.

3 Motion and likelihood models

The initial (reference) region surrounding the object of interest is chosen manually and is denoted as $\boldsymbol{t}_{ref}$. In our case this is a rectangular region, and we are tracking its centre. The model used for this region is given below.

3.1 Motion model

The motion of the moving object is modelled by the random walk model,

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}\boldsymbol{x}_k + \boldsymbol{v}_k, \quad (4)$$

with a state vector $\boldsymbol{x} = (x_k, y_k, s_k)^T$ comprising the pixel coordinates of the centre of the region surrounding the object, and the region scale s_k. $\boldsymbol{F}$ is the transition matrix ($\boldsymbol{F} = \boldsymbol{I}$ in the random walk model) and $\boldsymbol{v}_k$ is the process noise assumed to be white, Gaussian, with a covariance matrix $\boldsymbol{Q} = \mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_s^2)$. The estimation of the scale permits to adjust the region size of the moving objects, e.g., when it goes away from the camera, when it gets closer to it, or when the camera zoom varies.

3.2 Likelihood model

The normalised distance between the two regions $\boldsymbol{t}_{ref}$ (reference region) and $\boldsymbol{t}_{\boldsymbol{x}}$ (current region), for particle ℓ, is calculated according to (3), and then substituted into the likelihood function:

$$\mathcal{L}(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}^{(\ell)}) \propto \exp\left(-D^2(\boldsymbol{t}_{ref}, \boldsymbol{t}_{\boldsymbol{x}})/D_{min}^2\right), \quad (5)$$

where $\ell = 1, 2, \ldots, N$ and $D_{min} = \min_{\boldsymbol{x}}\{D(\boldsymbol{t}_{ref}, \boldsymbol{t}_{\boldsymbol{x}})\}$. This likelihood function is then used to evaluate the importance weights of the particle filter, to update the particles and finally the overall estimate of the centre of the current region $\boldsymbol{t}_{\boldsymbol{x}}$. Here $\boldsymbol{z}$ is a notation of the measurement vector, although with the SSIM we have no measurement in explicit form. We extract directly the structural properties of the region through the SSIM that are related to the estimates of the centre of the region of interest and we use directly the distance between the reference and current region.

4 A particle filter for object tracking

Particle filtering is a method relying on sample-based reconstruction of probability density functions. Multiple particles (samples) of the state are generated, each one associated with a weight which characterises the quality of a specific particle. An estimate of the variable of interest is obtained by the weighted sum of particles. Two major stages can be distinguished in the Particle Filter (PF) method: *prediction* and *update.* During prediction, each particle is modified according to the state model of the region of interest in the video frame, including the addition of random noise in order to simulate the effect of the noise on the state. In the update stage, each particle's weight is re-evaluated based on the new data. An inherent problem with particle filters is degeneracy (the case when only one particle has a significant weight). A *resampling* procedure helps to avoid degeneracy by eliminating particles with small weights and replicating the particles with larger weights. Various approaches for resampling have been

Table 1: The particle filter with structural similarity measure

Initialisation

1. for $\ell = 1, 2, \dots, N$, generate samples $\{\boldsymbol{x}_0^{(\ell)}\}$ from the initial distribution $p(\boldsymbol{x}_0)$. Initialise weights $W_0^{(\ell)} = 1/N$

For $k = 0, 1, \dots,$

Prediction Step

2. For $\ell = 1, \dots, N$, sample $\boldsymbol{x}_{k+1}^{(\ell)} \sim p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k^{(\ell)})$ from the motion model for the object region.

Measurement Update: evaluate the importance weights

3. The cue is used as "measurement". Compute the weights
$$W_{k+1}^{(\ell)} \propto W_k^{(\ell)} \mathcal{L}(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}^{(\ell)}). \tag{6}$$
based on the likelihood $\mathcal{L}(\boldsymbol{z}_{k+1}|\boldsymbol{x}_{k+1}^{(\ell)})$ (5) of the cue.
4. Normalise the weights, $\widehat{W}_{k+1}^{(\ell)} = W_{k+1}^{(\ell)} / \sum_{\ell=1}^{N} W_{k+1}^{(\ell)}$.

Output

5. The posterior mean state estimate $\boldsymbol{x}_{k+1}$ is computed using the collection of samples (particles)
$$\hat{\boldsymbol{x}}_{k+1} = \sum_{\ell=1}^{N} \widehat{W}_{k+1}^{(\ell)} \hat{\boldsymbol{x}}_{k+1}^{(\ell)}. \tag{7}$$

Selection step (resampling)

6. Multiply/ suppress samples $\boldsymbol{x}_{k+1}^{(\ell)}$ with high/ low importance weights $\widehat{W}_{k+1}^{(\ell)}$, in order to introduce variety and obtain N new random samples. The residual resampling algorithm described in [5, 10] is applied. This is a two step process making use of sampling-importance-resampling scheme.
* For $\ell = 1, 2, \dots, N$, set $W_k^{(\ell)} = \hat{W}_k^{(\ell)} = 1/N$.

proposed; for the work here the residual resampling method [5] was used.

The PF developed in this paper based on the similarity measure is given in Table 1.

5 Performance evaluation

The performance of our method is demonstrated over three video sequences, in which we aim at tracking a pre-selected moving person. The reference frames are shown in Figure 1. The first sequence, *cross*, originates from our database and contains three people walking quickly in front of a stationary camera. The main difficulties posed by this sequence are the colour similarity between the tracked object, the background and other passing people, and a temporal near-complete occlusion of the tracked person by a passer-by.

The second sequence used, *man*, has been obtained from [7]. It is a long recording showing a person walking along a car park. Apart from some similarities to the nearby cars, and the shadowed areas, the video contains numerous instabilities. These result from a shaking camera (changes in the camera pan and tilt), fast zoom-ins and zoom-outs, and a slightly altered view angle towards the end of the sequence.

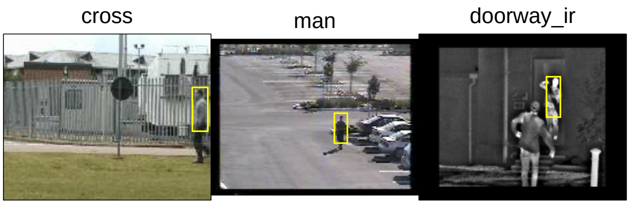

Figure 1: Reference frames from the test videos

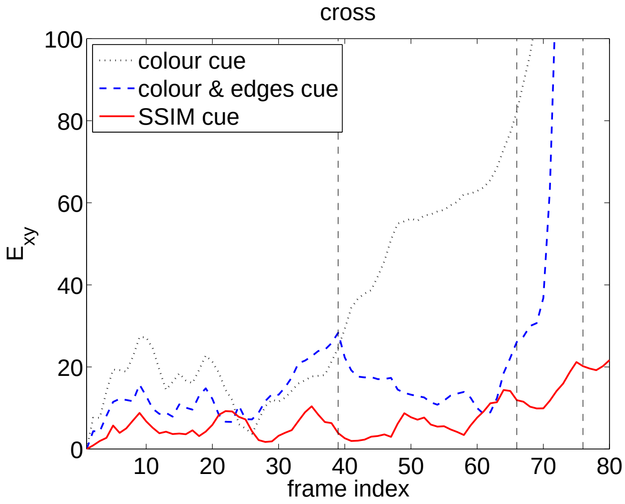

Figure 2: Plot of the RMSE of the object's central point for sequence *cross*. The frames marked by the vertical lines are given in Figure 5

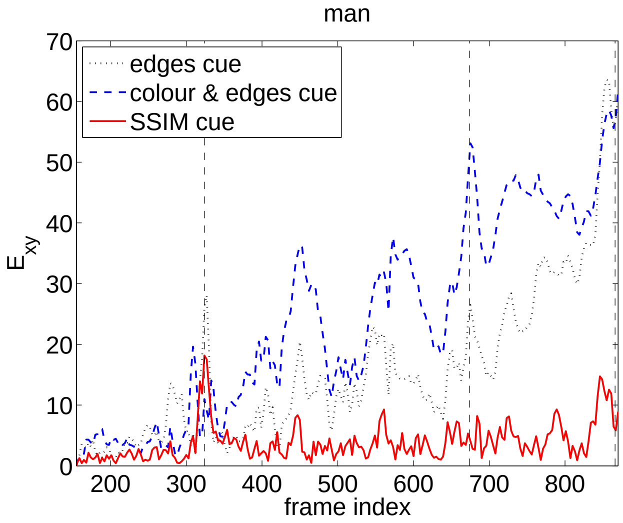

Figure 3: Plot of the RMSE of the object's central point for sequence *man*. The frames marked by the vertical lines are given in Figure 6

The third sequence, *doorway_ir*, being a part of our multimodal database, contains an infra-red recording of two people walking towards a stationary camera. The two persons look quite similar and the tracked object is often partially occluded by nearby objects.

In order to assess the performance of our tracking algorithm based on the similarity measure, we compare it with particle filtering tracking based on colour and edge cues proposed in [3]. The results presented below show that the PF with similarity measure outperforms the PF based on a single (colour or edge) and on fused (colour-and-edge) cue. In the PF based on fused cues, the likelihood is calculated as a product of the likelihoods of the separate cues as shown in [3].

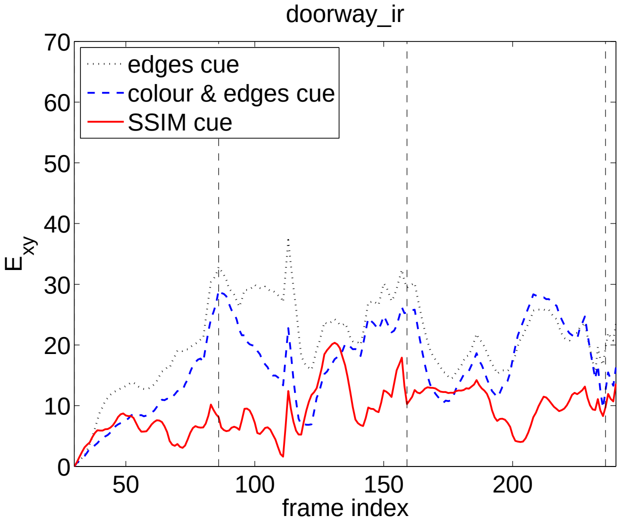

Figure 4: Plot of the RMSE of the object's central point for sequence *doorway_ir*. The frames are marked by the vertical lines are given in Figure 7

The model parameters are as follows: $\sigma_x = 2.5, \sigma_y = 10$, $\sigma_s = 0.01$ (for the *cross* sequence), $\sigma_x = \sigma_y = 2.5$, $\sigma_s = 0.05$, (for the *man* and *doorway_ir* sequence). The standard deviations of the noises are tuning parameters, although adaptations procedures are possible. This is an open issue that can be investigated in future, together with the necessity of finding an adaptive procedure for tuning the parameters of the SSIM, α, β and γ. Relatively low number $N = 100$ of particles has been used for all videos. The similarity measure has been calculated in the way proposed in [11], with $\alpha = \beta = \gamma = 1$.

The combined Root Mean Squared Error [2]

$$E_{xy}(i) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(x(i)-\hat{x}_m(i))^2 + (y(i)-\hat{y}_m(i))^2} \quad (8)$$

has been used to evaluate the performance of the developed technique. The pixel coordinates $(x(i), y(i))$ indicate the true position of the object and $(\hat{x}_m(i), \hat{y}_m(i))$ stand for estimated position in current frame i in $m = 1, 2, \ldots, M$ independent Monte Carlo realisations ($M = 50$ in our experiments). The manually created ground truth (the tracking box surrounding the object) has been used as the true coordinates.

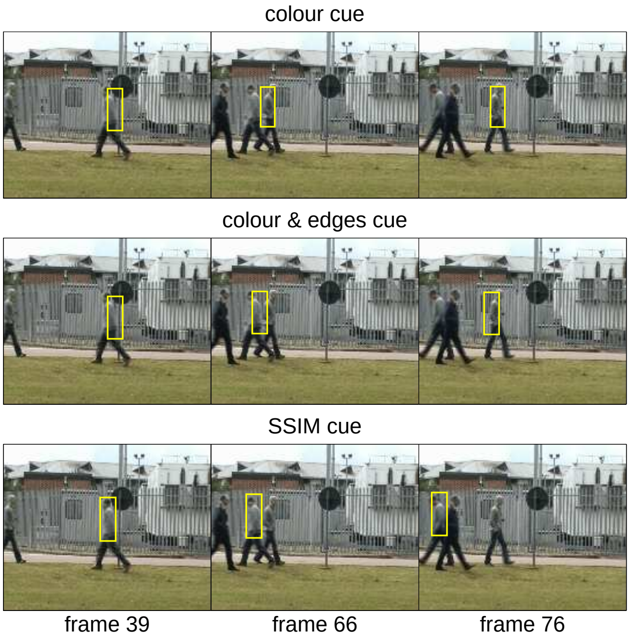

Figure 5: Frames with the tracker output superimposed, sequence *cross*

The error estimates are shown in Figure 2–4. Although all four described methods (based on colour, edges, colour-and-edges, and similarity measure) have been used, only the performance of the best three methods is shown in the plots for clarity. It can clearly be seen that the proposed method based on structural similarity, while never loosing the object, outperforms the other methods at nearly all instances.

A closer look at the selected output frames will illustrate the performance of different methods. Figures 5–7 show the object tracking boxes constructed from the mean locations and scales estimated during the tests.

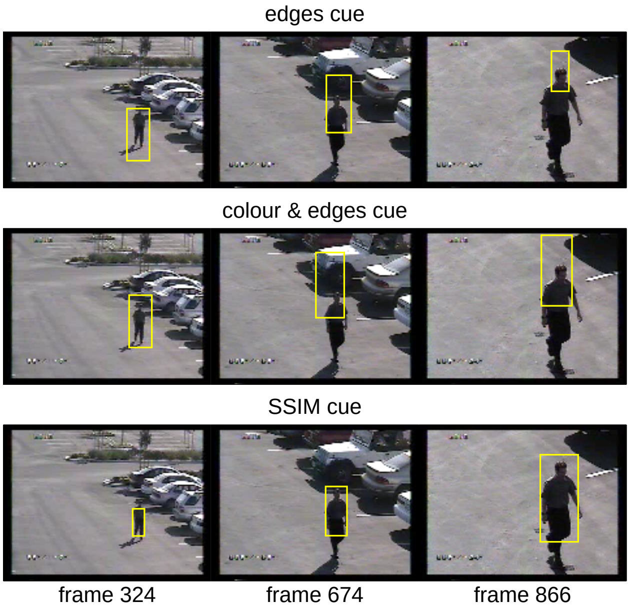

Figure 6: Frames with the tracker output superimposed, sequence *man*

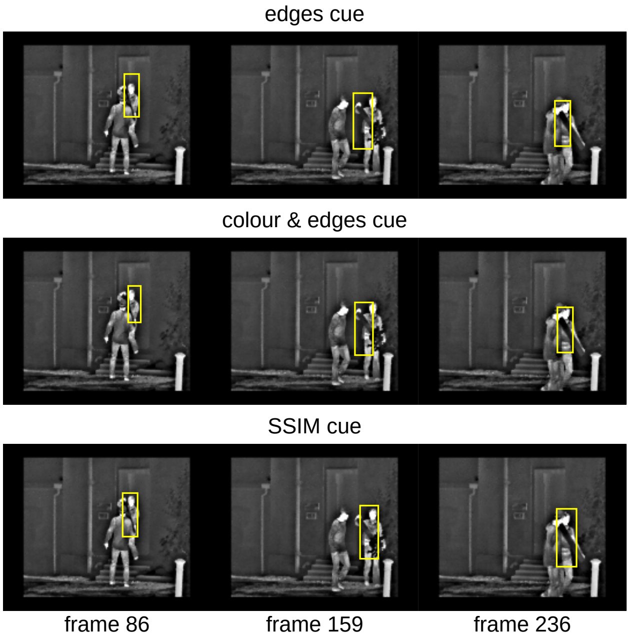

Figure 7: Frames with the tracker output superimposed, sequence *doorway_ir*

In the sequence *cross*, Figure 5, the first passer-by causes the colour and colour-and-edges cue tracker to loose the object (frames 66–76). Both trackers also seem to be attracted by the road sign (frame 39). The SSIM cue tracker is not distracted even by the temporary occlusion (frame 76).

The shaking camera in the sequence *man* (Figure 6, frame 324), introduces a small bias in the SSIM position estimate (while retaining correct scale), and the remaining trackers choose the wrong scale (whilst retaining the correct position). The two compared methods do not perform well in case of similar objects appearing close-by (shadow, tyre, frame 674) and rapid zoom of the camera (frame 866). Our method, however, seems to cope with both situations.

Although all the methods tested were able to track the person in the sequence *doorway_ir*, Figure 7, the proposed method is the most precise with respect both to position and correct scaling of the tracking box, for most frames in the video.

6 Conclusions

The new tracking scheme was tested with real-world video sequences and has been shown to perform reliably under different conditions. Colour cue itself cannot provide stable tracking under changing illumination and when there are regions with similar colour, such as those of the object. The fused colour-and-edge cue cannot provide a reliable tracking performance under ambiguous situations neither, especially with moving camera (with changes in the pan, tilt and zoom). The proposed particle filter based on the structural similarity measure shows the most stable and reliable performance. This is due to the fact that this measure captures the spatial similarity between the regions of interest, independently of the colour. It measures only relative changes in contrast and luminance which makes it more robust to the changes in the environment. The implemented tracking algorithm uses a changeable size of the tracking window, which makes it suitable for many real-world applications (where the camera–object distance varies significantly).

This paper presents early results obtained with this new method. Future work will be focussed on the extension of the presented method to achieve a degree of rotation invariance, and on theoretical justification of the results. The good performance of our methods when applied to both infrared and colour footage indicates that the structural similarity could be used in multimodal and fused video tracking. These predictions will also be verified by future investigation.

Acknowledgements

The authors are grateful to the financial support by the UK MOD Data and Information Fusion Defence Technology Centre, by projects 2.1 ‘Image and video sensor fusion’ and 2.2 ‘Communication optimisation for distributed sensor systems’.

References

[1] F. Aherne, N. Thacker, and P. Rockett. The Bhattacharyya metric as an absolute similarity measure for frequentcy coded data. *Kybernetica*, 32(4):1–7, 1997.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley and Sons, 2001.

[3] P. Brasnett, L. Mihaylova, N. Canagarajah, and D. Bull. Particle filtering with multiple cues for object tracking in video sequences. In *Proc. of SPIE's 17th Annual Symposium on Electronic Imaging, Science and Technology, V. 5685*, pages 430–441, 2005.

[4] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 25(5):564–577, 2003.

[5] J. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

[6] A. Łoza, T. D. Dixon, E. F. Canga, S. G. Nikolov, D. R. Bull, C. N. Canagarajah, J. M. Noyes, and T. Troscianko. Methods of fused image analysis and assessment. In *Proceedings of the Advanced Study Institute Conference, Albena, Bulgaria, 16–27 May (to appear)*, 2005.

[7] PerceptiVU, Inc. Target Tracking Movie Demos. http://www.perceptivu.com/MovieDemos.html.

[8] P. Pérez, J. Vermaak, and A. Blake. Data fusion for tracking with particles. *Proceedings of the IEEE*, 92(3):495–513, March 2004.

[9] G. Piella and H. Heijmans. A new quality metric for image fusion. In *Proceedings of the Intl. Conf. on Image Processing*, Barcelona, Spain, 2003.

[10] E. Wan and R. van der Merwe. The Unscented Kalman filter. In S. Haykin, editor, *Kalman Filtering and Neural Networks*, chapter 7, pages 221–280. Wiley Publishing, Sep. 2001.

[11] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, Apr. 2004.

[12] A. Webb. *Statistical Pattern Recognition.* John Wiley & Sons, 2003.