Region Template Correlation for FLIR Target Tracking

H.S. Parry[1], A.D. Marshall[1], and K.C. Markham[2]

[1]Department of Computer Science
University of Wales College of Cardiff
PO Box 916
Cardiff CF2 3XF

[2]British Aerospace (Dynamics) Ltd.
Flight Dynamics Department
PO Box 5, FPC 440,
Filton, Bristol BS12 7QW

E-mail: H.S.Parry@cs.cf.ac.uk, dave@cs.cf.ac.uk

Abstract

This paper deals with the problem of tracking an object from a sequence of images captured from a camera moving towards the object. Many conventional image based trackers trace key reference points over an image sequence using simple correlation processes [1]. However, the presence of noise and the magnification of image features as the camera approaches the object may cause a tracked point to drift. This paper introduces an alternative technique based on tracking regions formed by image segmentation. A region template of the target is acquired in the first frame and correlated over segmented images in future frames. The new technique produces significant reductions in drift rate and results have been obtained for real forward looking infrared (FLIR) images. This paper then discusses the advantages of integrating the new technique with a conventional point correlation based tracker.

1 Introduction

The tracking of features in an image sequence is important in the areas of robotics and automated navigation. The tracking process depends on matching a set of image features (*e.g.* points, edges or regions) present in one frame with features found in the next frame. Image points are typically matched using conventional correlation, where an image patch extracted from the previous frame is compared with a search area in the current frame [1, 2, 3]. Point correlation has the advantage of being robust and can be implemented for real-time applications. However, the problem associated with correlation is that, over a long image sequence, small tracking errors may accumulate to cause tracked points to migrate from their initially designated areas.

Our application for tracking image features is the guidance of a missile with an on-board FLIR camera towards a stationary object. For this purpose, only a subset of the motion parameters are required for navigation and these can be supplied from the use of a multiple point correlator(MPC) [1]. However, FLIR images are typically of low contrast and can worsen the drift of point correlation with the effect that background objects will become tracked. Therefore, there is a need for a method to correct drift throughout the missile's time-of-flight.

Tracking regions of segmented images [4, 5, 6] has the advantage of regions being a higher level representation than pixels. Their distinctive attributes make

BMVC 1996 doi:10.5244/C.10.38

them less prone to noise. Typically they are matched by size, shape and/or intensity. However, the segmentation of images into regions can be slow and a region may become divided across two frames leading to complications in the matching process. The division of regions is associated particularly in image sequences where the target enlarges due to egomotion[1].

This paper describes a new method of correlating a target region template across image frames that have been segmented and where the target has been selected by the *man-in-the-loop* in the first frame. This method of region template correlation (RTC) copes very well with regions which split between frames. This paper presents results of a typical test carried out on real FLIR image to show that RTC tracking can significantly reduce drift rates experienced by conventional correlation trackers. The reduction in drift results from the fact that regions offer a higher level representation of the image sequence evolution.

While this new method has been successfully run on its own, it is presented as a means of correcting drift experienced by the MPC. This paper describes the current method of tracking features used in our application and then the steps involved in the new method of tracking. The results for the current and new methods are compared. Finally, the integration of RTC with the MPC will be considered.

2 MPC Tracker

In our application, the current method of providing missile guidance employs the MPC tracker [1]. Each point is tracked by conventional correlation where a reference patch extracted from the previous frame is compared over a search area in the current frame to generate a correlation error surface. An extrema in the correlation error surface determines the image position that gives the best match in the current frame. The reference patch is then updated and centred on the correlation extrema. The scene similarity is provided by the difference squared correlator:

$$c(k,l) = \sum_{i=1}^{K_r} \sum_{j=1}^{L_r} (r(i,j,t-T) - s(i+k-1, j+l-1, t))^2$$

for: $k = 1, ..., K_s - K_r + 1$ and $l = 1, ..., L_s - L_r + 1$

where k and l are the offsets between reference image r and search space s.

The tracking of a point on an object whose shape, size or orientation changes across consecutive frames produces an error in the correlation. Although the error is small given a short time step between two consecutive frames, the accumulation of errors over a large number of frames cause tracked points to migrate from the target area [2, 3].

The tracking of multiple points allows the motion parameters of a moving camera to be estimated. For the purpose of guiding a missile or aircraft towards a stationary object only a subset of the motion parameters are required for navigation. The four dominant incremental guidance parameters are the pitch and yaw look angles to the target, a roll angle and an instantaneous estimate of time-to-go.

[1]Egomotion describes the type of image motion produced by a camera moving through a static environment.

The MPC tracker provides a good estimate of the required motion parameters between consecutive frames with improved accuracy as the number of tracked points increases. However, its inability to recover from tracking errors is a serious drawback and can lead to background objects being tracked. Since the MPC has been implemented in hardware and linked to the on-board navigation system there is a requirement for a method to periodically correct tracking errors.

3 Region Based Segmentation

The RTC tracker presented in this paper employs the region segmentation of an image. Region segmentation is the process of partitioning image pixels into regions of, for example, homogeneous grey values. A pixel becomes a member of a region on the conditions that its grey value falls within a specified range and that the pixel is spatially connected to another region member.

3.1 Segmentation Algorithm

The segmentation technique used in this application was based on histogram optimisation segmentation (HOS) [7]. This segmentation method was selected because it has been used successfully on FLIR images in the past [8] and our tests on real images have given good results considering that FLIR images are typically of poor contrast and have a high signal-to-noise-ratio. Furthermore, HOS has greater potential for real time implementation than traditional methods such as region growing that have not given significantly better results.

The HOS algorithm uses local histograms of an image, through the application of a 3x3 filter, to smooth regions of uniform intensity while enhancing edges. A clustering algorithm then scans the image line by line to group adjacent pixels that have an intensity difference below a segmentation threshold. An example of HOS image is given in Figure 1.

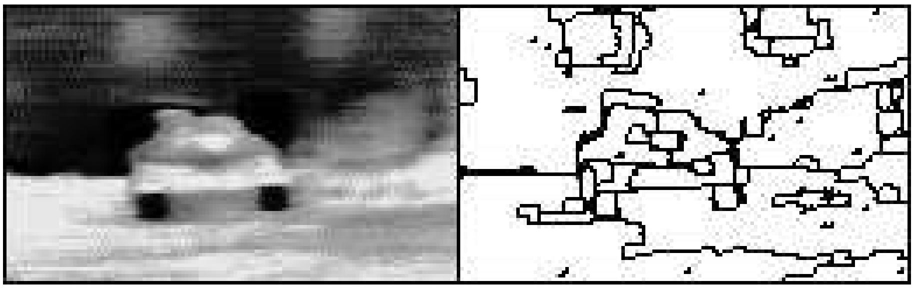

Figure 1: *Histogram optimised segmentation of a real FLIR image.*

4 Region Tracking

The segmentation of an image sequence leads to the need to match regions between frames so that objects may be tracked. The following sections describes previous work related to our application done on region tracking and the associated difficulty.

4.1 Target Tracking

The use of regions formed from segmentation has been investigated for target tracking [9, 10, 11]. These algorithms intended for real-time use, employed the geometric centroid of a segmented region as the target aimpoint. These were mainly intended to track missile or jet exhaust plumes and assumed that the target was small and had the strongest intensity. The methods made the assumption that the centroid position in the previous frame would fall on the target region in the current frame. For each frame, the image would be segmented into target and background regions within a window centred on the predicted aimpoint. No attempt was made to ensure a correspondence between regions in two consecutive frames.

These methods because of their sensitivity to target intensity are not valid when the target is large [12] or in a cluttered scene [10]. They fail because larger targets give rise to multiple regions and a cluttered scene does not allow the easy discrimination of a target from its surroundings.

4.2 Difficulty in Region Tracking

The work described above made no attempt to match regions between frames. The process of matching regions across a segmented image sequence captured from a camera undergoing egomotion can be very difficult. The complicating factors are the presence of noise and the fact that detail becomes more profound producing changes in image intensity as the distance to an object decreases. These factors affect the consistency of the segmentation over an image sequence and leave little correspondence between regions across two consecutive frames. While a region preserving its shape and size across frames is easy to relate, region splitting and merging complicate the matching process. Examples of methods that have coped with region splitting and merging have employed region corners [5] and multi-resolution segmentation [6].

The difficulty in tracking regions over an image sequence when there is little correlation between regions across two consecutive frames is illustrated in Figure 2. The two frames are segmentations of a battle tank in a cluttered scene. The two consecutive frames have different image partition and few regions in one frame can be be easily mapped onto the next frame. A region can change shape slightly, become joined with another, become divided into smaller region or a combination of the former two.

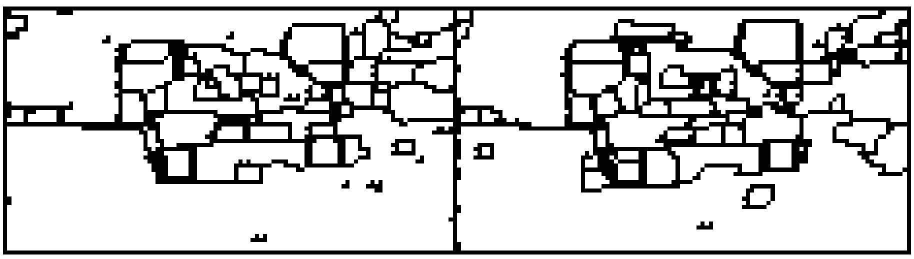

Figure 2: *Segmentation of two consecutive FLIR images.*

5 Region Template Correlation

As described in the previous section, the difficulty in tracking regions over an image sequence lies in the fact that there can be little correlation between regions across two consecutive frames. A method of tracking a target object over an image sequence needs to be able to cope with the unpredictable division of the target into multiple regions and also with target regions joined to the background. The latter problem is not as prevalent as the former in image sequences but nevertheless has the potential to be detrimental to the tracking of targets.

The solution to the problem of detecting the target when there is the unpredictable division of the target into multiple regions involves taking a template formed from target regions in the previous frame and finding regions in the next frame that best accommodate the template. The problem of a large proportion of the target being joined to the background affects the correct detection of the target. However, when this does occur its lifetime is short. Therefore, the problem can be overcome by detecting it and delaying the target position update until future frames.

5.1 Tracking Procedure

Our method uses a similarity function that evaluates how well the template fits in with underlying regions for a particular template position. The procedure to locate the target in the current frame is similar to the conventional correlation described in Section 2. A template representing the target area is passed from the previous frame and compared over a search window in the current segmented image frame. The search window is centred on the target position in the previous frame. For each point in the search window, the template is centred and its similarity with the segmented image[2] is measured. The point within the search window that gives the highest degree of similarity represents the target position in the current frame. Since one template is used for all target regions, the relative positions of the regions are rigidly maintained.

The similarity or correlation for a template position x, y and a segmented image is made according to the equation:

$$c(x,y) = \sum_{i=0}^{n} \frac{l_i^2}{r_i t}$$

where r_i is the area of the i^{th} region, t is the template area, l_i is the template overlap of the i^{th} region and n is the number of segmented regions.

The equation evaluates the correlation by measuring the proportion of overlap between the projected template and its underlying regions. Those regions which are totally or largely covered contribute a high value. The point in the search area giving the highest correlation corresponds to the point where the template successfully fits in with its underlying regions. Thus, where the template is acquired and correlated in the same frame (see Figure 3), under correct template alignment, all underlying regions would be 100 per cent covered giving a maximum correlation value of 1.0. When the template is acquired in one frame and correlated in the next frame, the peak correlation value will always be less than 1.0 because of the effect of noise and the magnification of the target.

A false target position may result from performing correlation on a poorly segmented image. The point of highest correlation does not correspond with the actual target position when a large proportion of the target is joined to a background region. This occurrence is signalled by the highest correlation value being

[2] While complete image segmentations are shown in this paper to give an idea of range, during tracking only a window around the target requires segmentation.

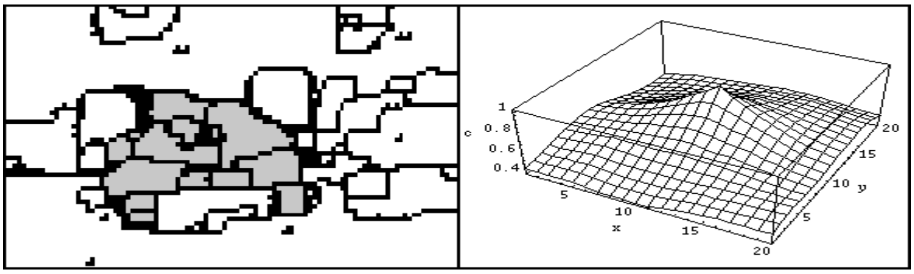

Figure 3: *Correlation surface generated from template acquired and correlated for segmented image on the left.*

Figure 4: *Segmentation of two consecutive images. The left image gives a high correlation for a template passed from previous frame, the right image gives a low correlation for the template highlighted in the left image.*

low and that it is unsafe to accept the returned position. This is illustrated in Figure 4. The frame on the left gives a correlation value of 0.96 and the template extracted for use in the next frame is highlighted. The frame on the right gives a correlation value of 0.48 for the reason that a target region is joined to the background. During a sequence run, correlation values below a *correlation threshold* results in the returned target position being rejected.

After the target position is found and providing the correlation value is above the specified *correlation threshold*, the template is updated to accommodate the increased target size and to obtain a better target definition. *Template update* is executed by projecting the template around the detected position of the target template and considering the overlap between the template and current segmented image. Regions whose proportion of overlap is above a specified *overlap threshold* are copied onto a new template. This step is necessary because even though the correlation value is high, small target regions may still be joined to the background. The *template update* process is not sensitive to minor misalignments between actual target and the template and can correct for errors in target position returned by template correlation. For target positions whose detection falls below the *correlation threshold*, the target position and template from the last update is passed to the next frame.

During *template update*, small target regions that are joined to the background will not be copied into the template and this can lead to a deterioration in the integrity of the template as cavities are left. To reduce this effect, the old template is aligned with the new to fill the cavities. This does not take into account an increase in the target size, and therefore target areas that are continually not updated eventually lead to targets parts not being represented by the template.

5.2 Results

The RTC tracker has been tested on real FLIR image sequences with encouraging results. The FLIR sequences were taken a decade ago and are considered to be of poor quality compared to today's standard FLIR cameras but are representative for present systems application when optics, motion blur, vibration and thermal noise are taken into account. Furthermore, the sequences provide an aggressive test of our tracking method. A typical example selected to illustrate the improvements is a sequence captured from a helicopter approaching a stationary tank. Figure 5 shows the segmented sequence with the template acquired for a particular frame superimposed on it and represented as a shaded area. Regions that are only partially covered by the template are the result of the old template being superimposed on the new template. For the first frame, the template was acquired by manually selecting target regions to be included.

Figure 6 is a plot of the correlation values against frame number obtained from the test sequence. The horizontal line represents the *correlation threshold* for the *template update*. Peak correlation values that fell below this line resulted in no *template update*. The fact that the plot oscillates severely during the initial stages reflects the poor image segmentation obtained due to the quality of the FLIR images. During poor segmentations a large target proportion is usually joined with the background. The target is composed of few regions during the early sequence stage and therefore a target region joined to the background represents a large target proportion. Consistent tracking was obtained once the range-to-go had halved.

To measure the drift rate of the MPC tracker, the distance of the centroid of the tracked points relative to the target centre was calculated. For the RTC tracker, the drift was measured as being the distance between the region template centre and the ideal target centre. From Figure 7 it can be seen that the drift for the MPC is more severe than the RTC tracker. By frame 115 the drift for the MPC is 17.8 pixels compared to 4.4 for the RTC, a significant improvement in navigation system performance terms.

5.3 Conclusion

The image sequence tested on the tracker was aggressive to ensure that under no circumstances would the tracker fail to recognise when a frame produces a bad segmentation and start tracking a false target. The segmentations are not good but this is a reflection of the quality of the images rather than the segmentation algorithm. Other segmentation methods based on region growing and split-and-merge produced only minor improvements compared to HOS but were significantly slower. Threshold steps were included to cope with the effects of poor segmentations. These steps controlled whether complete or partial update of the template took place. Failure of complete or partial update were mostly short lived and could be tolerated because the short time step between images meant that target size and position would change little. Therefore, the integrity of the template compared to the target was preserved.

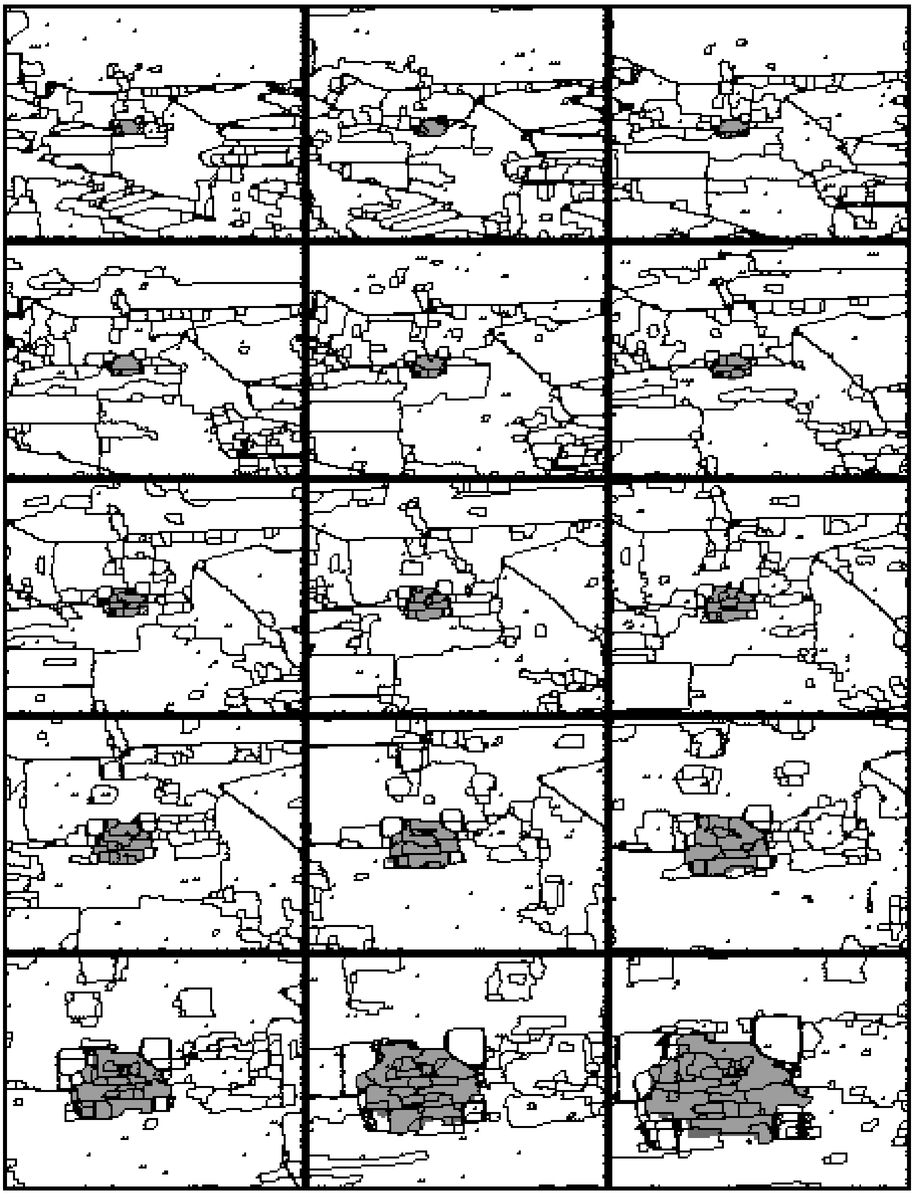

Figure 5: *Example of RTC applied to sequence of segmented real FLIR images. The shaded areas represent the template acquired for a frame.*

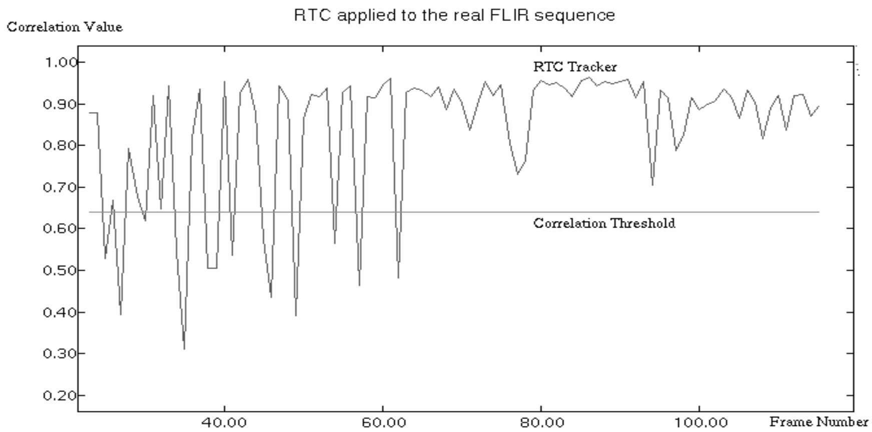

Figure 6: *Plot of correlation value against frame number. Correlation values below the correlation threshold are considered as false target detections.*

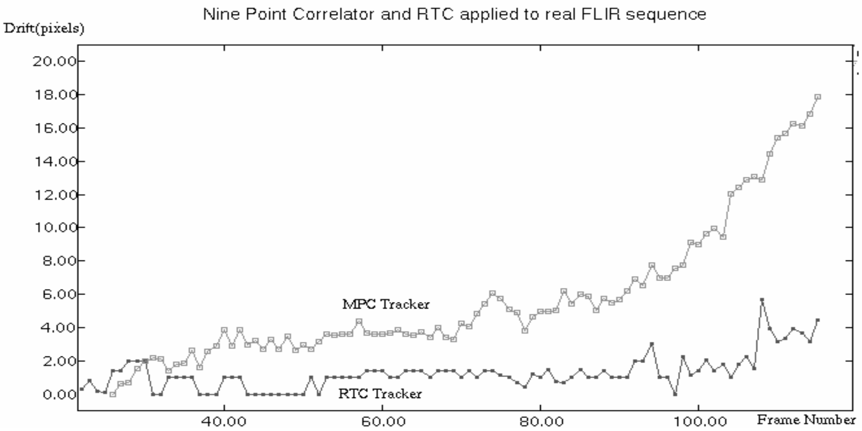

Figure 7: *Drift comparison of MPC and RTC trackers.*

6 Integration of MPC and RTC Trackers

The RTC tracker has been successfully executed on its own but can benefit from an estimate of the increase in target size made by the MPC tracker. RTC can take advantage of such an estimate to magnify its template and thus overcome two problems. The first problem occurs when a succession of peak correlation values falling below the *correlation threshold* results in a period of no *template update*. This leads to an impaired ability to detect the target because the template is smaller than the image target size. The second problem occurs during *template update* when the old template is superimposed on the new template. This can lead to a loss in template integrity when the new and old templates are of different sizes. The loss in target integrity may be seen in Figure 5 where the right track of the tank is included in the template during the early stage of the image sequence but is later not present.

The integration of both tracker types allows deficiency in one tracker type to be compensated by the other. However, for our application RTC is viewed as providing support for the MPC tracker for the following reasons. The MPC tracker has already been implemented in hardware and integrated with missile navigation systems but, as already explained, its main problem is the drift of tracked points form their initially designated areas. The execution of the RTC tracker on its own has a serious drawback for real time implementation and that is that both segmentation and correlation require increased processing as the target being tracked becomes larger. Therefore, for real time performance, the integration of the two tracker types may be done by periodically applying RTC after a certain number of frames to correct any drift experienced by the MPC tracker.

7 Conclusion

This paper has described a new method of tracking targets based on region template correlation. The new method tracks regions and employs a novel technique to cope with region splitting and merging between frames. It has been tested on real FLIR images and has demonstrated its superiority in comparison with the MPC tracker, the current tracking method. Further improvements may be obtained by the integration with the MPC tracker for real time implementation and is a topic of on-going research.

References

[1] K.C. Markham, "Time-To-Go Estimation from Infrared Images", IEE Proceedings I, **139**, 3, pp. 356-363, June 1992.

[2] D.A. Montera, S.K. Rogers, D.W. Ruck and M.E. Oxley, "Object Tracking Through Adaptive Correlation", Optical Engineering, **33**, 1, pp. 294-302, Jan. 1994.

[3] Q. Zheng and R. Chellappa, "Automatic Feature Point Extraction and Tracking in Image Sequences for Arbitrary Camera Motion", International Journal of Computer Vision, **15**, pp. 31-76, 1995.

[4] K. Price and R. Reddy, "Matching Segments of Images", IEEE PAMI, **1**, 1, pp. 110-118, Jan. 1979.

[5] Y.C. Kim and K. Price, "A Feature-Based Monocular Motion Analysis System Guided by Feedback Information", Proc. Image Understanding Workshop, DARPA, Morgan Kaufmann Publishers, pp. 611-620, 1993.

[6] G. Healey, "Hierarchical Segmentation-Based Approach to Motion Analysis", Image and Vision Computing, **11**, 9, pp. 570-576, Nov. 1993.

[7] G. Roberts, "Histogram Optimisation and Compression Segmentations", SPIE Applications of Imaging Technology Proc., **397**, pp. 487-493, April 1983.

[8] G.A. Roberts, "Model Guided Segmentation of Ground Targets", SPIE Applications of Artificial Intelligence V, **786**, pp.198-205, 1987.

[9] R.J. Schalkoff and E.S. McVey, "A Model and Tracking Algorithm for a Class of Video Targets", IEEE PAMI, **4**, 1, pp. 2-10, Jan. 1982.

[10] D.R. Van Rheeden and R.A. Jones, "Noise Effects on Centroid Tracker Aimpoint Estimation", IEEE AES, **24**, 2, pp. 177-184, March 1988.

[11] Y. Bar-Shalom, H.M. Shertukde and K.R. Pattipati, "Use of Measurements from an Imaging Sensor for Precision Target Tracking", IEEE AES, **25**, 6, pp. 863-871, Nov. 1989.

[12] E. Oron, A. Kumar and Y. Bar-Shalom, "Precision Tracking with Segmentation for Imaging Sensors", IEEE AES, **29**, 3, pp. 977-986, July 1993.