Position-, rotation-, scale-, and orientation-invariant multiple object recognition from cluttered scenes

Article (Accepted Version)

Position, rotation, scale and orientation invariant object tracking from cluttered scenes

Peter Bone, Rupert Young[1], Chris Chatwin

Laser and Photonic Systems Research Group

Department of Engineering and Design

University of Sussex,

Brighton BN1 9QT

ABSTRACT

A method of tracking objects in video sequences despite any kind of perspective distortion is demonstrated. Moving objects are initially segmented from the scene using a background subtraction method to minimize the search area of the filter. A variation on the Maximum Average Correlation Height (MACH) filter is used to create invariance to orientation while giving high tolerance to background clutter and noise. A log r-θ mapping is employed to give invariance to in-plane rotation and scale by transforming rotation and scale variations of the target object into vertical and horizontal shifts. The MACH filter is trained on the log r-θ map of the target for a range of orientations and applied sequentially over the regions of movement in successive video frames. Areas of movement producing a strong correlation response indicate an in-class target and can then be used to determine the position, in-plane rotation and scale of the target objects in the scene and track it over successive frames.

Keywords: tracking, logmap, MACH filter, correlation filter, invariant pattern recognition

1. INTRODUCTION

The ability to detect and track a particular object in a scene is useful for many practical applications. Since objects in a dynamic scene can move around, the system must be able to successfully detect objects despite variations in position, scale, rotation and orientation. It must also be able to detect objects in cluttered scenes if it is to be used in a practical application. To track target objects in video sequences, a system must first be designed to detect and correctly classify objects in individual video frames. An object can then be associated with the same object in previous frames if its position has remained within a certain proximity - or with the use of predictive matching. It will be assumed that the target objects in the scene will be moving. If this was not the case then a similar system for detecting objects in a static scene could be used. It will also be assumed that the camera is not moving. Using these assumptions, the process of searching individual video frames can be greatly optimised by only applying the detection filter to those parts of the image that are moving. The moving parts of the image can be extracted by keeping a running average image of the previous n frames and subtracting it from the current frame.

The problem of recognizing objects despite distortions in position, orientation and scale [1-2], and within cluttered backgrounds is a demanding pattern recognition problem. The first main success at solving the invariance problem came from the development of the Synthetic Disciminant Function (SDF) [3-5], which included the expected distortions in the

[1] E-mail: R.C.D.Young@sussex.ac.uk; Tel: +44(0)1273 678908; Fax: +44(0)1273690814

filter design to create invariance to such distortions. More recent attempts have been based on the Maximum Average Correlation Height (MACH) filter [6], which can be tuned to give maximum performance and is far more immune to background clutter. To attempt to solve the full invariance problem, we combine two existing techniques, each capable of achieving invariance to several of the possible variations of a target object. Out-of-plane rotation invariance is achieved using a Maximum Average Correlation Height filter (MACH). Aside from creating out-of-plane rotation invariance, this filter is capable of discriminating the target objects from cluttered or noisy backgrounds. Scale and in-plane rotation invariance is created with the use of a log r-θ mapping (logmap) of a localised region of the image space. A change in scale or rotation in the target object results in a horizontal or vertical shift in the logmap, which makes the object detectable by correlation with the logmap of the reference image.

2. IN-PLANE ROTATION AND SCALE INVARIANCE

To detect target objects in a scene despite their differences in scale or in-plane rotation to the target reference images a log r-θ mapping [7-9], or logmap, can be employed [10]. The logmap uses a variation on the basic x-y grid sensor used in conventional image processing. The structure of the sensor is based on a Weiman polar exponential grid [7,11-13] and consists of concentric exponentially spaced rings of pixels, which increase in size from the centre to the edge. This produces an arrangement similar to that found in the mammalian retina where photoreceptive cells are small and densely packed in the fovea and increase in size exponentially to create a blurred periphery. Each sensor pixel on the circular region of the x-y Cartesian space is mapped into a rectangular region in Polar image space r-θ. The sensor's geometry maps concentric circles in the Cartesian space into vertical lines in the polar space and radial lines in the Cartesian space into horizontal lines in the Polar space. This transformation offers scale and rotation invariance about its centre, since rotation or scale changes simply produce vertical or horizontal shifts in the polar space. Fig. 2.1 shows the complex logarithmic mapping performed by the sensor geometry. The vertical lines in the *w*-plane map to concentric circles in the *z*-plane and the horizontal lines in the *w*-plane map to radial lines in the *z*-plane.

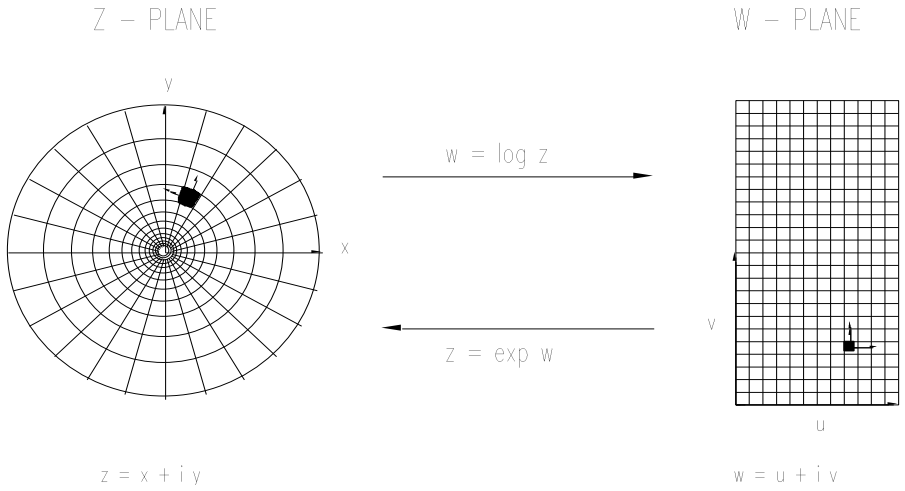
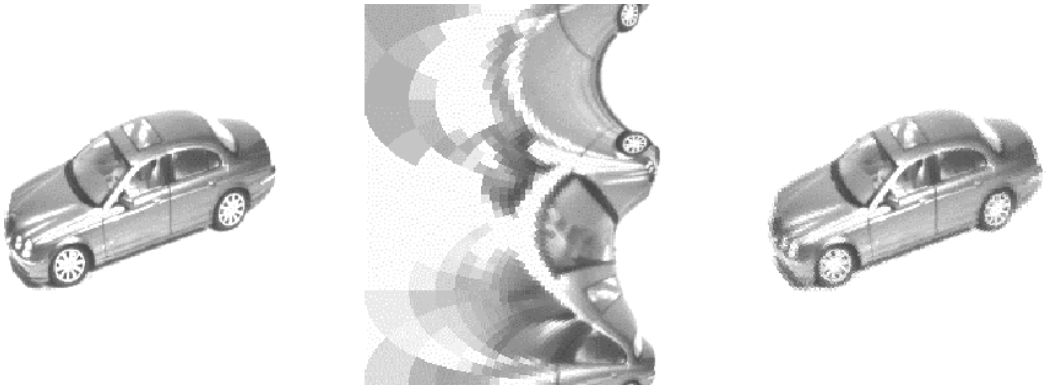


Fig. 2.1 Sensor geometry and logarithmic mapping

The fact that rotation and scale changes result in horizontal and vertical shifts in the logmap means that the logmap creates invariance to these transformations since a linear correlator object recognition system is invariant to x-y translation. It should be emphasised however, that the log-polar mapping is not shift invariant, so the properties described above hold only if they are with respect to the origin of the Cartesian image space.

Figure 2.2 shows an original image (a), the log-mapped image (b) and the inverse mapping of the logmap (c).



(a) (b) (c)

Fig. 2.2 Original image (a), logmap image (b) and inverse mapping (c)

3. MAXIMUM AVERAGE CORRELATION HEIGHT (MACH) FILTER

The MACH filter [6], like the SDF, is another method of creating invariance to distortions in the target object by including the expected distortions in the construction of the filter. The MACH filter maximizes the relative height of the average correlation peak with respect to the expected distortions. Unlike the SDF, the MACH filter can be tuned to maximize the correlation peak height, peak sharpness and noise suppression while also being tolerant to distortions in the target object that fall between the distortions given in the training set. However, the peak height of the MACH filter is unconstrained making it more difficult to interpret the results of the correlation[14].

The optimal trade-off (OT) MACH filter (in the frequency domain) is given as [15]:

$$h = \frac{m_x^*}{\alpha C + \beta D_x + \gamma S_x} \tag{3.1}$$

where α, β and γ are non-negative OT parameters, m_x is the average of the training image vector $x_1, x_2, \ldots, x_N$ (in the frequency domain), and C is the diagonal power spectral density matrix of additive input noise. It is usually set as the white noise covariance matrix, $C = \sigma^2 I$. D_x is the diagonal average power spectral density of the training images:

$$D_x = \frac{1}{N}\sum_{i=1}^{N} X_i^* X_i \tag{3.2}$$

where X_i is diagonal matrix of the i^{th} training image. S_x denotes the similarity matrix of the training images:

$$S_x = \frac{1}{N}\sum_{i=1}^{N} (X_i - M_x)^* (X_i - M_x) \tag{3.3}$$

where M_x is the average of X_i.

The different values of α, β and γ control the MACH filter's behaviour to match different application requirements [16]. If $\beta=\gamma=0$, the resulting filter behaves much like a MVSDF filter [17] with relatively good noise tolerance but broad peaks. If $\alpha=\gamma=0$ then the filter behaves more like a MACE filter [18], which generally exhibits sharp peaks and

good clutter suppression but is very sensitive to distortion of the target object. If $\alpha=\beta=0$, the filter gives high tolerance for distortion but is less discriminating.

4. FULLY INVARIANT TRACKING SYSTEM

By combining the in-plane rotation invariance of the logmap and the distortion invariance of the MACH filter it was possible to create a filter that is invariant to any kind of geometrical distortion of the target object, while maintaining high performance even in cluttered scenes. Such a filter was constructed and tested for the problem of tracking a particular car (a model of a Jaguar S-Type) in a dynamic scene (Fig 4.1). The system was expected to correctly detect the car in every frame despite variations in its out-of-plane orientation, in-plane rotation, scale, position and with noisy or cluttered backgrounds. To create invariance to changes in out-of-plane orientation of the car, the MACH filter was created using a training image set consisting of the expected range of rotation taken at small intervals of viewing angle.

Fig. 4.1 Various distortions of the target object (a model of a Jaguar S-Type)

The problem of combining the logmap with the MACH filter was solved simply by creating a logmap of each reference image before synthesis of the filter. The input image then also needed to be log-mapped before being correlated with the filter. However, the logmap gains in-plane rotation and scale invariance at the expense of position invariance. This means that to search the entire frame for target objects requires the correlation process to be repeated for each location where the target could be found. The size of this search space can be greatly reduced by firstly extracting the parts of the scene that are changing – since we have assumed that the target object is moving and the camera is not.

A moving average image was calculated from the previous n frames and subtracted from the current frame under inspection. Resulting pixels that were close to zero could then be assumed to be part of the background since they had not changed. Pixels greater than a given threshold could be assumed to have changed and must therefore be part of the moving objects. The number of previous frames used to construct the moving average needed to be set at an optimum value to give the best results. If it was set too low then moving objects in previous frames would not be adequately averaged out and would cause objects to be repeated in successive frames. If it was set too high then the average would not adapt quickly to changes in the background such as lighting conditions. The number of frames for creation of the average was set to 20, although this would need to be changed depending on the expected speeds of moving objects in the scene. Single disconnected pixels that had been classed as moving were considered noise and removed to reduce the number of separate objects. A dilation operation was also implemented to reduce the chance of single objects being classed as multiple separated objects due to noise. The regions of movement were then grouped using a morphological labelling method and region centres were found to locate the centre of each moving object.

To classify each region as an in-class target or out of class object, the MACH logmap filter was applied to each region in turn. A 128 by 128 sub-image, centred over each region, was logmapped and correlated with the MACH filter. The correlation peak height of the sub-image with compared to a threshold - above which the sub-image was classified as containing an in-class target. The threshold was calculated by correlating each of the reference images with the MACH filter and taking an average of their resultant peak intensities (the intensity at the centre of the correlation plane). This value then gave a value close to what would be expected after correlation between the filter and an in-class target. The threshold was then set slightly lower than this value by multiplying by 0.9 so that any in-class objects would produce peaks slightly above the threshold, allowing for some reduction in amplitude, while still being high enough for all out-of-class objects to fall below the threshold. The threshold equation was thus calculated as:

$$Threshold = \frac{0.9}{N}\sum_{i=1}^{N} CentrePeak\left(FFT(h) * FFT(t_i)\right) \quad (4.1)$$

Where h is the MACH filter created using equation 3.2. t_i is the logmap of the i^{th} training image.

It was found that object centres found by background subtraction did not always correspond to the exact object centre required for a successful detection using the filter. Therefore, the filter did not always perform successfully when applied exclusively at the centre of the object. This was because the background subtraction method did not always find the entire object and the training images used to construct the MACH filter were not centred exactly. To resolve this problem the filter was also applied around the near proximity of the centre. The area around the centre was raster scanned over a range of 4 pixels around the object centre in 2 pixel increments and the filter was applied at each position. The largest correlation peak height found over this range was taken as the peak height for that object and was compared to the threshold for classification.

Once a target object had been detected it is then possible to calculate the in-plane rotation and scale of the target in the image. This is possible since rotation and scale variations in the target object produce horizontal and vertical shifts in the logmap. The position of the maximum correlation peak will therefore also be shifted from its central position when target objects are rotated or scaled relative to the set of reference images used to build the filter. By measuring the horizontal and vertical offset of the peak from the centre of the correlation plane it is possible to calculate the scale ratio and rotation relative to the reference images. From equations 2.2 and 2.9 it can be seen that the rotation and scale can be calculated as:

$$ScaleRatio = \exp\left(u_{offset} \cdot \mathrm{g}\right) \quad (4.2)$$

$$Rotation = v_{offset} \cdot \mathrm{g} \quad (4.3)$$

Where u_{offset} and v_{offset} are the horizontal and vertical offsets of the maximum correlation peak from the centre of the correlation plane.

As well as calculating the scale and in-plane rotation of detected targets, it is also possible to calculate the out-of-plane rotation (orientation) as a post processing operation. This is done by correlating the log-mapped sub-image with each of the original log-mapped reference images and seeing which one gives the strongest response.

The parameters used in the MACH filter function were set to α=0.01, β=0.3 and γ=0.1. These values were set after a long period of testing with different values to correctly balance the discriminating ability against the sharpness of the correlation peak and the general performance of the filter.

The system was able to process approximately 10 sub-images per second, including logmap creation, two Fourier transforms, correlation peak location and classification, running on a 3GHz Pentium using simulation code written in MATLAB.

4.1 Results

A video sequence of the Jaguar S-Type model was created using a mini DV digital camcorder. Initially, the system was tested using a single moving target. As an initial test, the model was moved across the floor in a way that maintained its orientation from the point of view of the camera. It was not necessary to sample every frame of the video since the model was moved relatively slowly – so every 7^{th} frame was sampled to speed up processing. The brightness and contrast of the frame images were altered to enhance the range of intensities across the surface of the objects. The MACH logmap filter was constructed using 1 training image at a similar orientation to that of the model in the video sequence (0 degrees rotation with an elevation angle of 30 degrees). The system performed well and classified the model as a target in every frame. This was despite the fact that the video image was of slightly lower quality than the training images, which had been taken using a high resolution digital still camera, due to video compression and interlacing effects. The lighting used in the filming of the video sequences also made it different to that of the training images, but the filter was able to cope well with this due in part to the edge enhancing nature of the MACH filter.

The target car was filmed again but this time moving in such a way that its orientation changed from the point of view of the camera. This is a relatively severe test since the filter is constructed from training images of the model at discrete orientation intervals, whereas in the video the model is changing orientation smoothly. This means that the filter must detect the target despite its orientation being at an intermediate angle relative to the training images. The filter was constructed using training images at 0, 5 and 10 degrees of orientation corresponding to the range of out of plane rotation of the model in the video sequence. The system performed well again with strong, sharp peaks in the correlation plane which were always greater than that of the detection threshold calculated using equation 4.1. This demonstrates that the filter shows good invariance to orientation (out of plane rotation) even when the orientation is at an intermediate angle to the images used to build the MACH filter.

To test the discrimination ability of the filter, a different model car was filmed as an out of class object. The two cars were filmed simultaneously to test the systems ability to cope with multiple moving objects and show that the system can correctly classify both objects using the same system variables. The target car was again filmed changing orientation slightly and moving slightly towards the camera (changing scale). The filter was constructed using training images at 5, 10 and 15 degrees of out of plane rotation. The system performed well and correctly classified the target model and out of class model in every frame. Figure 4.2 shows frame samples 15 to 18 (top row), the corresponding moving parts of the scene found using background subtraction (second row), the correlation intensity plane for the out of class object (third row) and the correlation intensity plane for the target object (fourth row). The boxes around the objects show the regions of movement and their classification as in class or out of class (dark or light colour respectively). It should be noted that the position of the correlation peak corresponds to the rotation and scale of the target and not the position of the target in the scene since the correlation is performed in logmap space. It can be seen that the lower out of class object has sections missing in the moving parts of the scene images. This was due to reflections from the lights creating a similar intensity to that of the floor. This did not cause any problems since the boundary of the object remained in-tact and the object centre is found by calculating the centre of the region's bounding rectangle. If this had caused a problem then it could have been improved by finding the moving parts of the scene from the colour frames before converting them to greyscale for filtering. Figure 4.3 shows an enlarged correlation intensity graph for a typical in class target (taken from frame sample 19 from the same video sequence). At this point in the video the target model was at a slightly larger scale than that of the training images used to construct the filter, which demonstrates the scale invariance of the filter created by the log-mapping.

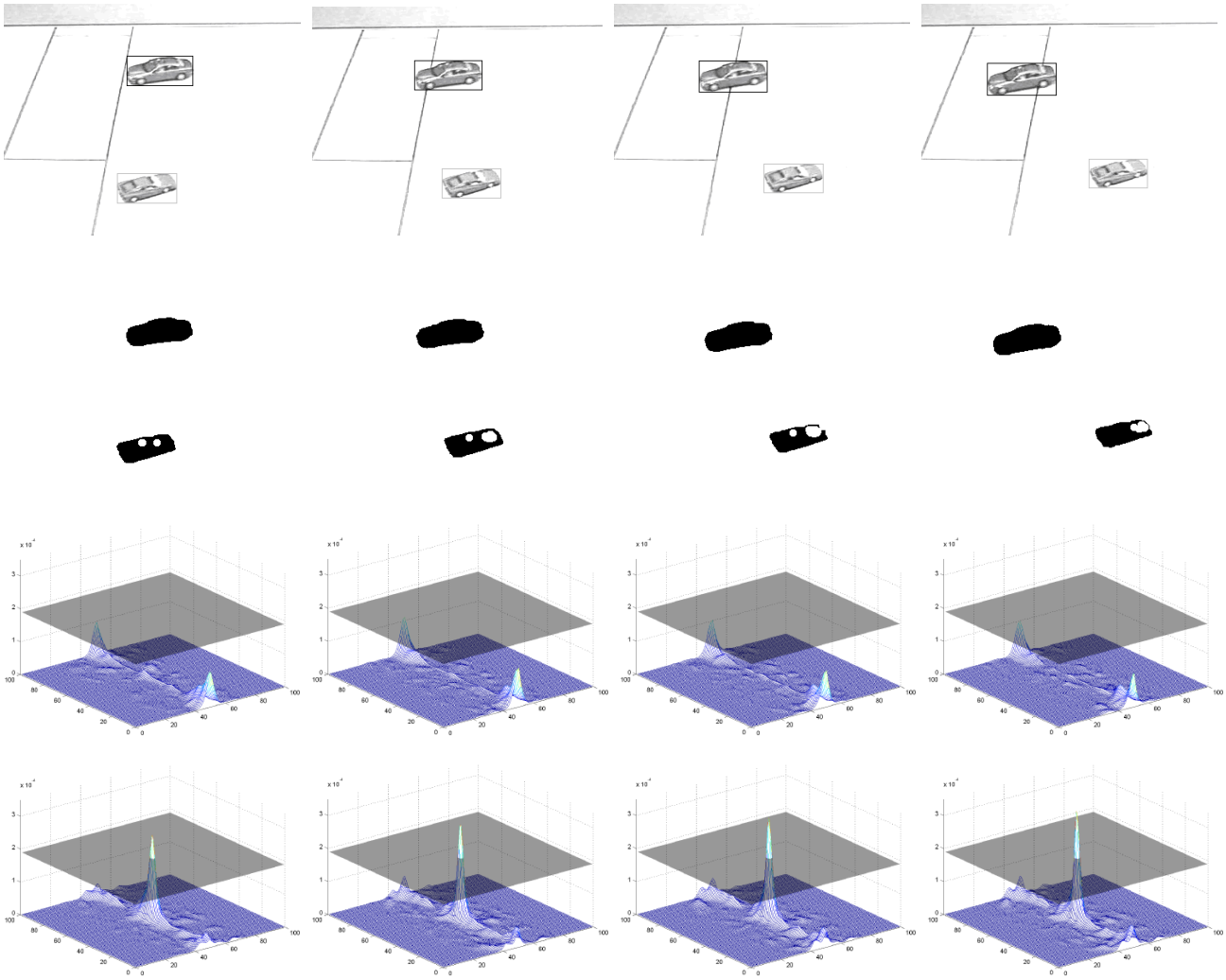


Fig. 4.2 Four consecutive frames of the video. Row 1 shows the original frame images. Row 2 shows the parts of the scene that are moving. Row 3 shows the correlation intensity of the out of class object (the lower car) and Row 4 shows the correlation intensity for the target object (the upper car). The transparent plane shows the detection threshold.

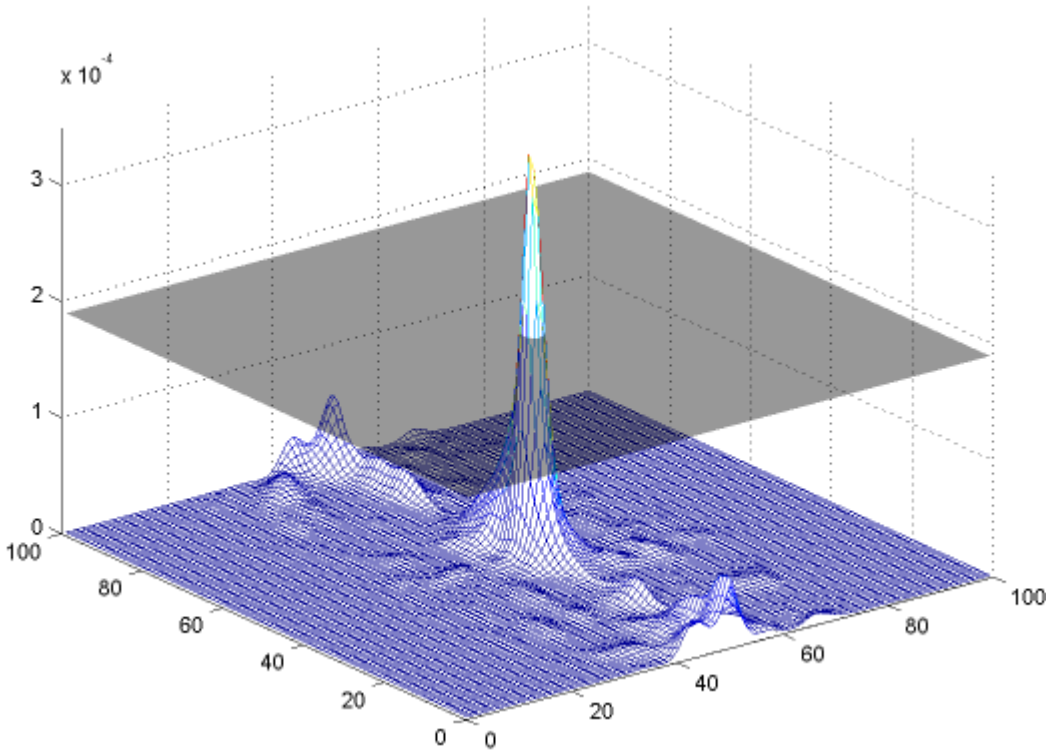


Fig. 4.3 A typical correlation intensity plane for a target object. The transparent plane shows the detection threshold calculated using equation 4.1

5. CONCLUSIONS

By combining a MACH filter with a log-mapping method it was possible to create a filter capable of detecting a target object in a cluttered scene despite any kind of geometrical distortion. By finding the moving parts of the scene in a video sequence by background subtraction and applying the filter to each part in successive frames it was possible to track a target object as it moved. A system such as this is useful for many practical applications where target objects are moving in a way that varies their position, rotation, scale or orientation. The implementation of a MACH filter gave high performance by giving tolerance to background clutter and noise while providing invariance to orientation of the target object. A method of classification of correlation peaks as in-class or out-of-class objects was employed by computing the average expected peak intensity, achieved by correlating the filter with each of the reference images. This made object detection simple since the height of the peak could directly be compared to a predefined threshold.

Invariance to in-plane rotation and scale of the target was successfully achieved by log r-θ mapping the training set images and input image prior to synthesis of the filter and correlation. Since the filter was only applied to the parts of the scene that were moving it was able to process each frame at fairly high speed – although it was slowed down slightly by the fact that the filter had to be applied several times in the proximity of the object in order to find its true centre. With the use of optimised code and a fast machine it would be possible to implement this system in its current state as software. However, since the filter is based on Fourier techniques it would be possible to implement the system using optical hardware by employing a Spatial Light Modulator (SLM) to sample the moving parts of the frame images at high speed and an optical correlator to filter the sampled objects and produce the correlation output. It may also be possible to implement the system in real time using a specialized Digital Signal Processing (DSP) chip set and associated efficient code. This would reduce the processing time further if needed – for example, if targets are moving at high speed and a higher frame sampling rate is required.

Further improvements to the system could be made by implementing predictive tracking such as that demonstrated by the Kalman filter [19]. This would mean that multiple target objects could be tracked in the scene simultaneously with greater reliability. It would also solve the common problem of maintaining tracking as target objects obscure each other as they pass.

REFERENCES

1 D. Casasent and D. Psaltis, "Position, rotation, and scale invariant optical correlation", *Applied Optics* Vol. **15**, No. 7, 1795-1799 (1976)

2 K. Mersereau and G. Morris, "Scale, rotation, and shift invariant image recognition", *Applied Optics* Vol. **25**, No. 14, 2338-2342 (1986)

3 H. J. Caulfield and W. Maloney, "Improved discrimination in optical character recognition", *Applied Optics, Vol.* **8**, *No.* **11**, 2354-2356 (1969)

4 C. F. Hester and D. Casasent, "Multivariant technique for multiclass pattern recognition", *Applied Optics*, Vol. **19**, 1758-1761 (1980)

5 Z. Bahri and B. V. K. Kumar, "Generalized synthetic discriminant functions", *Journal of Optical Society of America, Vol.* **5**, *No.* 4, 562-571 (1988)

6 A. Mahalanobis, B.V.K. Vijaya Kumar, S. Song, S.R.F. Sims, J.F. Epperson, "Unconstrained correlation filters", *Applied Opt*ics, Vol. **33**, pp. 3751-3759, (1994)

7 Weiman, C.F.R., and Chaikin, G., "Logarithmic spiral grids for image processing and display," *Computer Graphics and Image Processing* Vol. **11**, 197-226 (1979)

8 Sandini, G., and Dario, P., "Active vision based on space-variant sensing," in: *5th International symposium on Robotics Research*, 75-83 (1989)

9 Schwartz, E.L., Greve D., and Bonmasser, G., "Space-variant active vision: definition, overview and examples," *Neural Networks*, Vol. **8**, No. 7/8, 1297-1308 (1995)

10 P.Bone, R. C. D. Young, C. Chatwin, "Position, rotation, scale and orientation invariant multiple object recognition from cluttered scenes", Accepted for publication, *Optical Engineering.*

11 Weiman, C.F.R., "3-D sensing with polar exponential sensor arrays," in: *Digital and Optical Shape Representation and Pattern Recognition*, Proc. SPIE Conf. on Pattern Recognition and Signal Processing, Vol. **938**, 78-87 (1988)

12 Weiman, C.F.R., "Exponential sensor array geometry and simulation," in: *Digital and Optical Shape Representation and Pattern Recognition*, Proc. SPIE Conf. on Pattern Recognition and Signal Processing, Vol. **938**, 129-137 (1988)

13 C-G. Ho, R.C.D. Young, C.R. Chatwin, "Sensor Geometry and Sampling Methods for Space-Variant Image Processing", *Journal of Pattern Analysis and Applications*, Vol. **5**, 369-384, (2002)

14 I. Kypraios, R. C. D. Young, P. Birch, C. Chatwin, "Object recognition within cluttered scenes employing a Hybrid Optical Neural Network (HONN) filter", *Optical Engineering*, Special Issue on Trends in Pattern Recognition, Vol. **43**, No. 8, 1839-1850, (2004)

15 Ph. Refregier, "Optimal trade-off filters for noise robustness, sharpness of the correlation peak and Horner efficiency", *Optics Letters*, Vol. **16**, No. 11, 829-831 (1991)

16 Hanying Zhou and Tien-Hsin Chao, "MACH filter synthesising for detecting targets in cluttered environment for gray-scale optical correlator", Proc. SPIE, Vol. **715**, 394-398, (1999)

17 B. V. K. Kumar, "Minimum Variance Synthetic Discriminant Function", Journal of Optical Society of America A3, Vol. **3**, 1579-1584, (1986)

18 A. Mahalanobis, B. V. K. Vijaya Kumar, D. Casasent, "Minimum average correlation energy filters", *Applied Optics*, Vol. **26**, No. 17, 3633-3640, (1987)

19 Brown, R. G., Hurang, P. Y. C., "Introduction to Random Signals and Applied Kalman Filtering", 2nd Ed, *John Wiley & Sons*, (1992)