

Data and Decision Level Fusion of Temporal Information for Automatic Target Recognition

K. Messer and J. Kittler

University of Surrey

Guildford Surrey GU2 5XH United Kingdom

K.Messer@surrey.ac.uk, J.Kittler@surrey.ac.uk

Abstract

Automatic Target Recognition (ATR) is a demanding application that requires separation of targets from a noisy background in a sequence of images. In our previous work [5] the background was adaptively described using two-dimensional filters designed by Principle Component Analysis on sampled two-dimensional image patches. Significant improvements in performance have been obtained by decision level fusion over time. In this paper we extend this idea and utilise the temporal nature of the data further to design a set of three-dimensional texture filters based on randomly sampled three-dimensional image patches. We show that by virtue of data level fusion, using these new filters the true-positive rate can be increased further whilst reducing the number of false-positives.

1 Introduction

Automatic Target Recognition (ATR) is concerned with the detection, tracking and recognition of small targets using input data obtained from a multitude of sensor types such as forward looking infrared (FLIR), synthetic aperture radar (SAR) and laser radar (LADAR). Applications of ATR are numerous and include the assessment of battlefield situations, monitoring of possible targets over land, sea and air and the re-evaluation of target position during unmanned missiles weapon firing.

An ideal system will exhibit the properties of a low false positive rate (detection of a non-target as a target), whilst obtaining a high true positive rate (the detection of a true target). This performance should be invariant to the following parameters: sensor noise; time of day; weather types; target size/aspect and background scenery. It should be flexible such that it has the ability to detect previously unseen targets and be able to retrain itself if necessary. It is unlikely that one single system will cope well with all these possible scenarios [2]. The many challenges produced by ATR have been previously well documented in [3], [8] and [1].

In this paper an adaptive ATR system is proposed, which decides how to best distinguish the target from a particular background or clutter. In the bootstrap phase a statistical model of the background is built by using a set of texture filters. In operation, the same features are computed for each new pixel output by the sensor input. A statistical test is then applied to this pixel feature vector to determine whether it belongs to the same region as the background or it is an outlier, i.e. a potential target. This general background/target concept is not new with some systems already using such an approach [6].

The novelty of this work is in the techniques applied to obtain a suitable set of filters which ensure that the background/target separation is maximised during training. In our previous work [5] we demonstrated that the use of a set of adaptive texture filters to model each background outperformed the more traditional Wavelet-based feature extractor. This set of filters was designed using Principal Component Analysis on randomly sampled image patches taken from a training image. This ensured that these filters had a mean response when presented with a similar looking texture. If an object with different texture, such as a target, is presented to the filter the resulting response should be non-mean, making its detection as an outlier easier. In this paper the filter design methodology is enhanced further to take into account the temporal dimension of the image data, i.e. the PCA is used to build 3-dimensional texture filters. These filters which perform data level fusion are shown to suppress the detection of false positives further.

The rest of this paper is organised as follows: in the next section the target detection algorithm is detailed in full. In section 3 the filter generation methods of PCA are explained whilst section 4 details the model optimisation step. Next we describe our tracking algorithm before results of our experiments on two image sequences are given. Finally, some conclusions are drawn.

2 Target Detection

In this paper the target detection problem is viewed as an outlier detection problem. In otherwords, anything that does not normally occur in the background is viewed as a potential target. The target detection algorithm proposed has three basic steps

Model Generation The homogeneous background is described using a basic statistical model.

Model Optimisation The model and model size are optimised using training data (if available).

Target Detection Outliers are found by deciding, per pixel, whether it is accurately described by the model.

In order to model the background, a feature vector for every pixel in the background areas of the training image is computed, $\mathbf{f}_i = [y_0, y_1, \dots, y_n]$ where $1 \leq i \leq N$ and N is the total number of pixels in the background area. Each y_k represents a measurement obtained by the k^{th} filter where $1 \leq k \leq n$.

The distribution of the N feature vectors is assumed to be normal for the homogeneous textured region of image and can be described by its mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ using

$$p(\mathbf{f}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp \left[-\frac{1}{2} (\mathbf{f} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{f} - \boldsymbol{\mu}) \right] \quad (1)$$

To detect targets in test frames the same set of n filters are used to generate a feature vector for every test pixel from that frame. Each feature vector, \mathbf{f}_{test} , is then tested in turn to see whether it belongs to the same distribution as the background or is an outlier (i.e. possible target). This is done using a measure known as the Mahalanobis distance:

$$d_M = (\mathbf{f}_{test} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{f}_{test} - \boldsymbol{\mu}) \quad (2)$$

The magnitude of d_M determines how probable it is that the corresponding pixel is a target. By selecting a threshold d_{thr} we can determine when to consider a pixel a target.

3 Filter Design

The aim of this work is to describe the background adaptively using Principal Component Analysis (PCA, also known as the Karhunen-Loeve transform). It is an extension of our previous work in [5] in which we compared PCA methods against a standard Wavelet-based method and a method based on Independent Component Analysis. It was concluded that the PCA method performed best. In our previous work the filter design was two-dimensional. In this paper we incorporate the temporal dimension into the filter design.

3.1 Two-Dimensional Filters

Principal Component Analysis [4] finds a linear base to describe the dataset. It finds axes which retain the maximum amount of variance in the data. To construct a PCA base, firstly N random rectangles of size $r \times c$ are taken from a set of training images. These rectangles are then packed into a $r \times c$ -dimensional vector \mathbf{x}_i , usually in a row-by-row fashion. This results in a data set \mathbf{X} containing N samples. Assuming that the global mean of the vectors in \mathbf{X} is zero, the principal components are the eigenvectors of the covariance matrix $\mathbf{X}\mathbf{X}^T$. These are the columns of the matrix \mathbf{E} , satisfying

$$\mathbf{E}\mathbf{D}\mathbf{E}^{-1} = \mathbf{X}\mathbf{X}^T \quad (3)$$

where \mathbf{D} is a diagonal matrix containing the eigenvalues corresponding to the eigenvectors in \mathbf{E} . The set of 2D filters is then generated by unpacking each row of \mathbf{E}^T into a filter of size $r \times c$.

3.2 Three-Dimensional Filters

The design of the 3D filters follows the same process as for the 2D design, however instead of extracting image rectangles from a single image, the rectangles are taken from d consecutive images. This data is then unpacked to form a vector of $(r \times c \times d)$ dimensions. Typically, d is set to 3. Figure 1 shows how one such data sample is typically constructed from a 3D image patch.

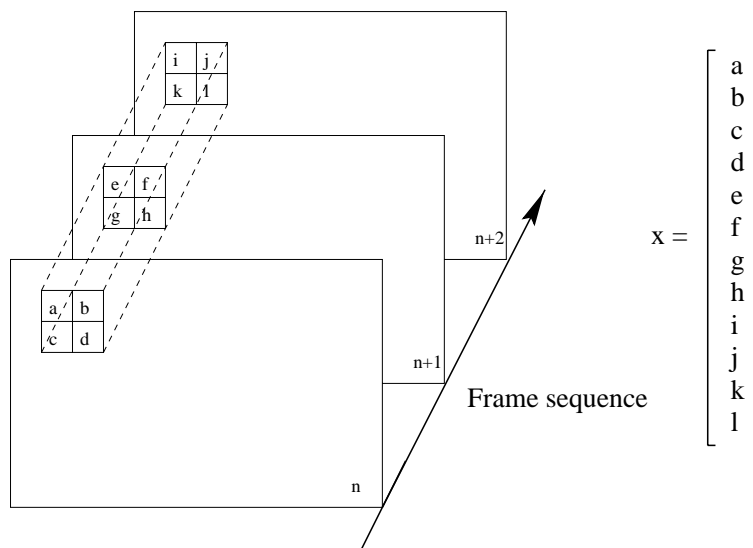


Figure 1: Generation of a single 12-dimensional data sample x from a 3D image patch, where $r = 2$, $c = 2$ and $d = 3$. Data set X is constructed by extracting 2000 random image patches.

4 Model Optimisation

In the above approach PCA generates many different texture filters which have different responses as each has been tuned to the different image features present in the training frame. For example, one filter may be sensitive to horizontal edges whereas another may be sensitive to vertical edges. Also, in general, the first few filters (i.e. the ones with the corresponding highest eigenvalues), try to capture the maximum intensity spread within the image. Clearly, the utility of the output of these filters for target recognition will be limited, as typically the targets lie in the middle of the intensity histogram.

To solve these problems a model optimisation stage is added which attempts to find a subset of these filters which are tuned better to solving our target/background separation problem. This optimisation process is performed using the *sequential forward selection* algorithm [7]. This algorithm provides a near-optimal solution and will find a set of features which maximises the Mahalanobis distance between background and target. More information on this model optimisation step can be found in [5].

Figure 2 shows an example of two such filters selected by this process.

5 Temporal Tracking

A real target will be persistent across a sequence of frames whilst genuine noise is random and should disappear after just a few frames. This information is utilised in a post-processing step which rewards targets that appear in approximately the same location across a series of consecutive frames.

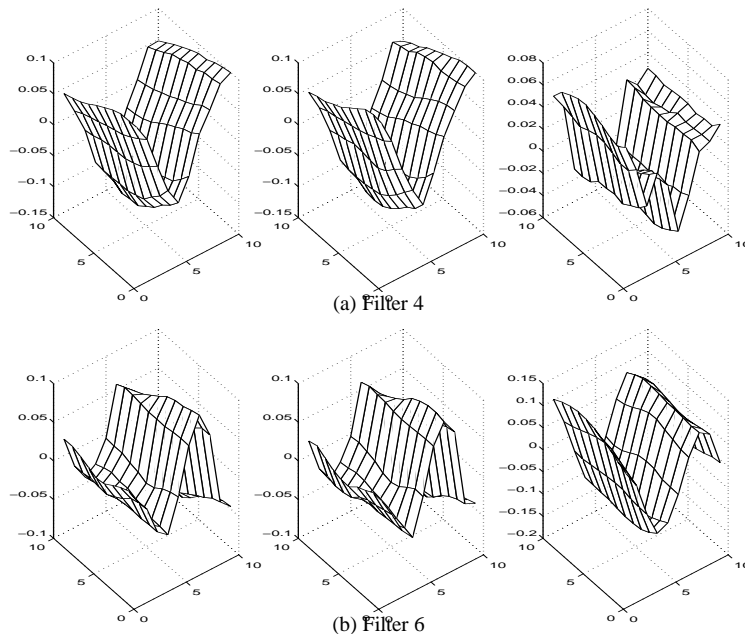


Figure 2: The two best filters (from 300), with dimensions $(10 \times 10 \times 3)$, chosen in the model optimisation stage.

Each thresholded target image is first dilated using a 3×3 mask (this allows for small movement of the target between each frame). Next, t consecutive dilated images are added together and the resultant image linearly scaled between 0 and 255. This resulting image displays how likely and persistent each target is. The brighter the blob the more certain that the target found is a real threat. Low intensity blobs can be dismissed as background noise. A second threshold could be applied at this stage if desired.

For this application it is important that the target be detected as quickly as possible. For this reason the temporal average of the results is made over only five frame period (i.e. a response time of one-fifth of a second).

6 Experiments

The proposed target detection technique has been applied to several sequences made available by DERA Farnborough. Typical results are shown in this section on a simulated sequence, SEASIM, and on a real sequence, AM.

6.1 On SEASIM

This sequence contains about twenty frames which have been artificially generated using a standard ray-tracing package. It represents the scenario of a sensor attached to a ship looking out over the ocean. Figure 3(a) shows the first frame of this sequence. Five targets have been inserted into this sequence; whose locations are given by the ground

	2D-PCA	3D-PCA
True Positives	4.19	4.69
False Positives	18.06	4.00

Table 1: Table showing average TP and FP rates for the SEASIM sequence

truth image of figure 3(b). These targets are very small (typically one pixel) and represent missiles moving towards the observer. Finding targets in this sequence is extremely hard, even for human observers.

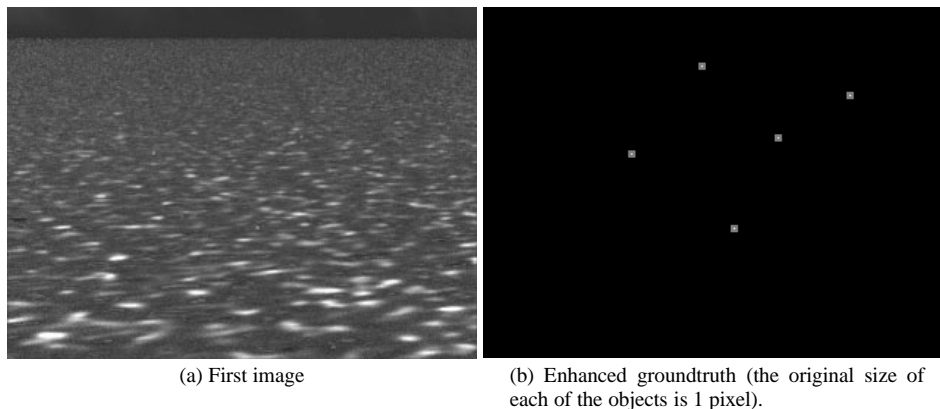


Figure 3: Sequence SEASIM.

The average true-positive (TP) count without temporal averaging, using 2D-PCA, is (4.19). For 3D-PCA this rises to (4.69). More interestingly the average false-positive rates (FP) for 2D-PCA is (18.06) which decrease dramatically for 3D-PCA, (4.00). A plot of FP versus frame number for this sequence is given in figure 7(a).

Figure 4 shows the temporally averaged results whilst using 2D-PCA. The corresponding results using the 3D filters are given in figure 5. Notice how the TP rate is now at a maximum in both sequences and the number of FP's is lower than the average values given above. The false-positives have also been given a lower intensity value.

On analysis of the nature of the false positives it was observed that most occur at the locations of breaking waves. It should be possible to dismiss many of these by using a longer temporal averaging mask, i.e. more than five frames. However, this time lapse should also be set as low possible to allow for the earliest possible warning.

6.2 The AM sequence

A real infra-red sequence of just 8 frames was acquired. Five targets (representing missiles) of varying maximum intensities were placed in the sea area. The first frame of this sequence along with the ground truth image is shown in figure 6. Again manual identification of these targets is extremely difficult.

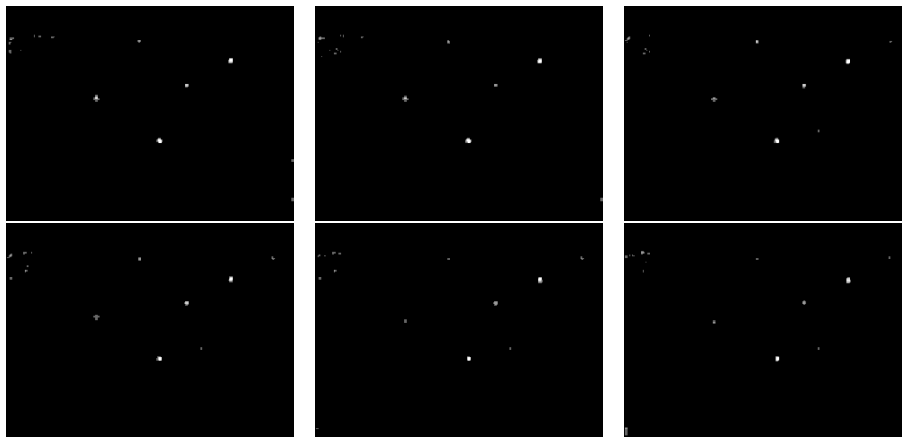


Figure 4: SEASIM: 2D-PCA - temporally averaged results.

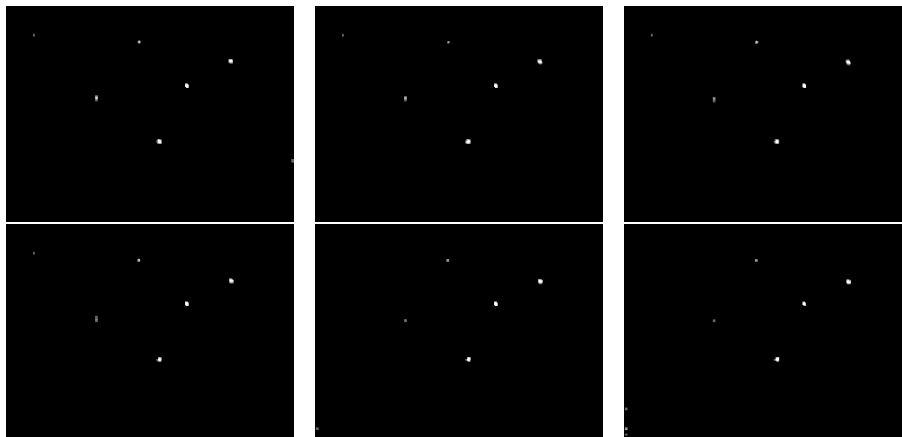


Figure 5: SEASIM: 3D-PCA - temporally averaged results.

	2D-PCA	3D-PCA
True Positives	5.00	5.00
False Positives	14.29	4.86

Table 2: Table showing average TP and FP rates for the AM sequence

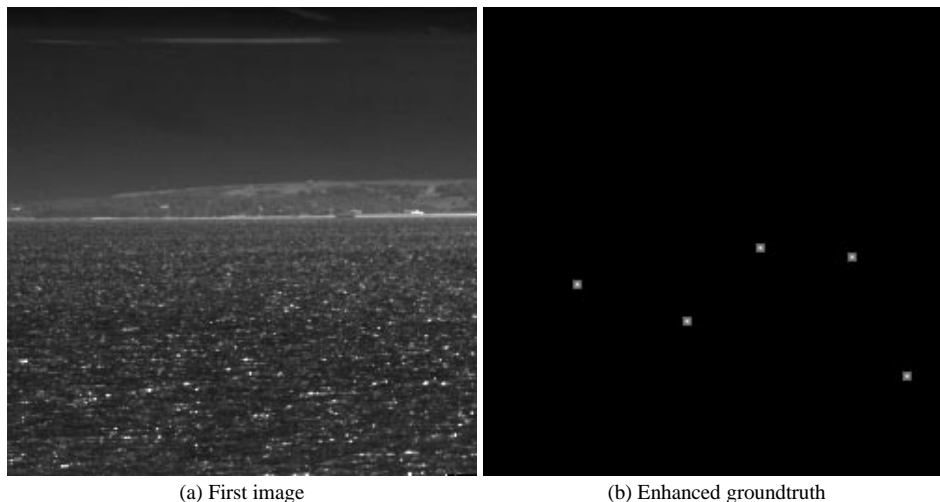


Figure 6: Sequence AM.

This time the average true-positive count without temporal averaging, using both 2D-PCA and 3D-PCA was at a maximum, i.e. (5.00). Also, the average false-positive rate for 2D-PCA was again higher than 3D-PCA, (14.29) and (4.86) respectively. The results of the temporal averaging are given in figure 8 for 2D-PCA and figure 9 for 3D-PCA. Again, notice how the actual targets appear brighter than the false-positives and the number of false positives has been reduced (especially for 2D-PCA).

7 Conclusions

In this paper we have demonstrated how the use of temporal information can provide us with a more robust automatic target recognition system. We have shown that by incorporating the temporal nature of the data into our PCA-based filter design, the corresponding features describe the background texture in a more homogeneous way. A slightly higher true-positive rate and a significantly lower false-positive rate, when using these 3D-filters, was observed.

The post-processing temporal averaging step eliminated some of the rogue false-positives as noise whilst enhancing the spatially and temporally consistent outliers.

Unfortunately, the system still has a number of free parameters which are set manually, such as the filter dimensions and outlier threshold. To make an operational system these either need to be fixed or set automatically. This is still an outstanding research

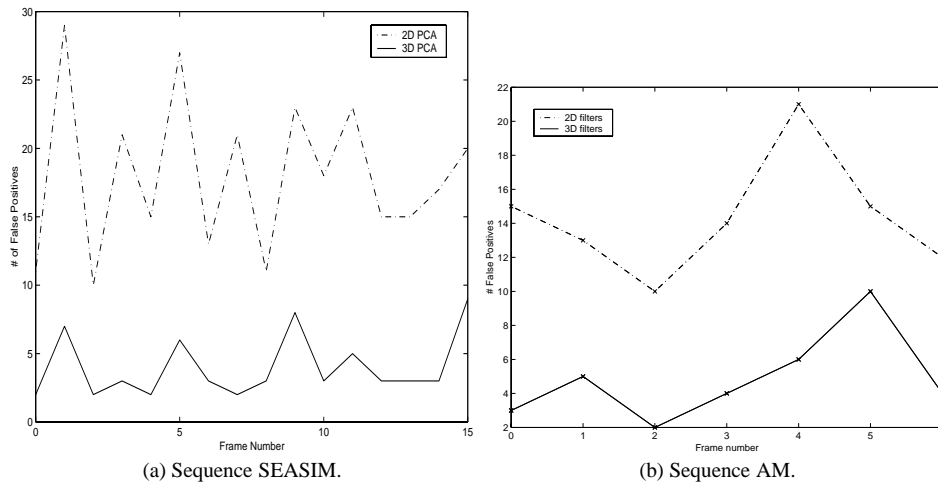


Figure 7: Plots of false positives.



Figure 8: AM: 2D-PCA - temporally averaged results.



Figure 9: AM: 3D-PCA - temporally averaged results

problem.

The statistical model we have chosen for the background in this paper is the simple uni-modal Gaussian. A problem with this approach is that the distribution of the features, even for a homogeneous region of image, may not be normally distributed. We are in the process of investigating more complex models such as Gaussian mixture models. Preliminary results indicate that the actual performance increase is fairly small and that this extra modelling comes at an extra significant computational cost.

The method of outlier detection can also be improved upon. At present our systems treats all points with a low probability density function value as a target. In reality the targets occupy only a corner of low density probability space. By using a support vector machine in a boosting phase one can attempt to classify just the outliers as target or background. Support Vector Machines are ideal for this type of application because the amount of training data available in this application is very low for one of the classes (i.e. the targets). Initial results are very promising. This approach also eliminates the need for a user to set a target threshold.

Acknowledgements This research was funded by the MoD under the Corporate Research Program by Technology Group 10: Information Processing and Technology Group 3: Aerodynamics, Propulsion, Guidance and Control. ©British Crown copyright 2000. Published with the permission of the Defence Evaluation and Research Agency on behalf of the Controller of HMSO.

References

- [1] B Bhanu. Automatic target recognition: State of the art survey. *IEEE Transactions on Aerospace and Electronic Systems*, 4:364–379, July 1986.
- [2] B Bhanu, D Dudgeon, E Zelnio, A Rosenfeld, D Casasent, and I Reed. Introduction to the special issue on automatic target detection and recognition. *IEEE Transactions on Image Processing*, 6:1:1–3, January 1997.
- [3] B Bhanu and T Jones. Image understanding research for automatic target recognition. *IEEE Transactions on Aerospace and Electronic Systems*, pages 15–22, October 1993.
- [4] P Devijver and J Kittler. *Pattern Recognition: A Statistical Approach*. Prentice Hall, 1982.
- [5] K Messer, D. de Ridder, and J Kittler. Adaptive texture representation methods for automatic target recognition. In *Proc British Machine Vision Conference BMVC99*, September 1999.
- [6] M.R. Moya and D.R. Hush. Network constraints and multi-objective optimization for one-class classification. *Neural Networks*, 9(3):463–474, 1996.
- [7] P Pudil, J Novovicova, and J Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, 15:1119–1125, 1994.
- [8] M Roth. Survey of neural network technology for automatic target recognition. *IEEE Transactions on Neural Networks*, 1:28–43, March 1990.