

Review

Trends in Correlation-Based Pattern Recognition and Tracking in Forward-Looking Infrared Imagery

Mohammad S. Alam ^{1,*} and Sharif M. A. Bhuiyan ²

¹ Department of Electrical and Computer Engineering, University of South, Alabama Mobile, AL 36688-0002, USA

² Department of Electrical Engineering, Tuskegee University, Tuskegee, AL 36088, USA; E-Mail: bhuiyans@mytu.tuskegee.edu

* Author to whom correspondence should be addressed; E-Mail: malam@southalabama.edu; Tel.: +1-251-460-6117.

Received: 18 June 2014; in revised form: 16 July 2014 / Accepted: 16 July 2014 /

Published: 24 July 2014

Abstract: In this paper, we review the recent trends and advancements on correlation-based pattern recognition and tracking in forward-looking infrared (FLIR) imagery. In particular, we discuss matched filter-based correlation techniques for target detection and tracking which are widely used for various real time applications. We analyze and present test results involving recently reported matched filters such as the maximum average correlation height (MACH) filter and its variants, and distance classifier correlation filter (DCCF) and its variants. Test results are presented for both single/multiple target detection and tracking using various real-life FLIR image sequences.

Keywords: patter recognition; target tracking; forward looking infra-red (flir) imagery; correlation filter

1. Introduction

Pattern recognition deals with the detection and identification of a desired pattern or target in an unknown input scene, which may or may not contain the target, and the determination of the spatial location of any target present. In pattern recognition or classification, the input is an image while the output is a decision signal based on some characteristic features of the input image. The number of features is usually fewer than the total necessary to describe the complete target of interest, and this

leads to a loss of information. Because of the crucial role of decision making required in pattern recognition, it is fundamentally an information reduction process, whereby, it is not possible to reconstruct the pattern but it is possible to give a precise decision [1–3].

Although a great deal of effort has been expended on detecting objects in visual images, only limited amount of work has been reported on the detection and tracking of targets in infrared images. In general, existing methods on infrared images work for a limited number of situations due to various practical constraints. An infrared sensor detects infrared radiation and converts it to an image by converting the temperature difference between an object and the surrounding background. The temperature scale is converted into a color scale or a gray scale on a display and in this way an image is obtained. This type of sensor or camera can image an object through smoke in a burning house, heat leaking from a house or objects in the absence of any reflected light (at night) [4]. The images captured by infrared sensors are becoming an integral part of the ongoing research on automatic target recognition (ATR).

Forward-looking infrared (FLIR) images are frequently used in ATR applications. The detection and discrimination of targets in infrared imagery has been a challenging problem due to low signal-to-noise ratio (SNR) and the variability of target and clutter signatures. The FLIR sequences, tested in this work, are recorded from a moving platform and include independently moving objects under various distortions and background variations. Thus, sensor ego-motion and object motions induce coupled motions into the FLIR images, which make the detection and tracking of the objects extremely complicated. To detect independently moving objects in FLIR image sequences, the sensor properties must also be taken into account.

Real life FLIR imagery demonstrates a number of well-known challenges such as significantly high level of variability of target thermal signatures, size/aspect, locations within the scene; large number of target classes; lack of prior information; obscured targets; competing cluttered background scenery; different geographic, meteorological and weather conditions; time of the day; high ego-motion; sensor noise; and variations caused by translation, rotation, and scaling of the targets. Furthermore, inconsistencies in the signature of targets, similarities between the signatures of different targets, limited training and testing data, camouflaged targets, non-repeatability of target signatures, and difficulty in exploiting contextual information make the recognition problem even more challenging in target detection in FLIR imagery. In the case of FLIR images, additional challenges are caused due to the following important differences [5–7] with visual sequences:

- The thermal images are obtained by sensing the radiation in the infrared spectrum, which is either emitted or reflected by the object in the scene. Due to this property the images obtained from an infrared sensor have extremely low SNR, which results in limited information for performing detection or tracking task.
- FLIR imagery smoothes out object edges and corners leading to a reduction of distinct features.
- The generation and maintenance of kinetic energy usually heats up a moving object (e.g., friction, engine combustion). Consequently, moving objects often appear brighter than the background.
- FLIR images are noisy and have less contrast. Moreover, they often contain dirt on the lens, or local sensor failure at certain pixel locations.

- FLIR sequences are not easily available (especially not from controlled experiments) and have a lower resolution. The sequences available to us are 128×128 pixels as compared to the 512×512 pixels, and more, of standard visual cameras.
- FLIR sequences are often under difficult circumstances and may have abrupt discontinuities in motion.

Due to the limitations and difficulties of FLIR imagery, they demand more robust techniques than visual sequences. This paper presents some widely used pattern recognition and target tracking techniques adopted for FLIR imagery. In this paper, we discuss several target detection and tracking algorithms which are based on the recently reported matched filter-based correlation techniques such as the MACH, EMACH, DCCF, and PDCCF filters. The performance of these algorithms was tested using real life FLIR image sequences supplied by the Army Missile Command.

2. Matched Filter-Based Correlation

The matched filter-based correlator was first introduced in 1964 [8]. In this technique, the input signal $f(\sigma, \varepsilon)$ is Fourier transformed to yield

$$F(o, v) = \mathfrak{F}[f(\sigma, \varepsilon)] \quad (1)$$

where \mathfrak{F} represents the Fourier transform operation, σ and ε represents the spatial domain variables, and o and v represents the frequency domain variables, respectively. The correlation output is obtained by inverse Fourier transform operation given by

$$g(\sigma, \varepsilon) = \mathfrak{F}^{-1}[F(o, v)H(o, v)] \quad (2)$$

where, \mathfrak{F}^{-1} represents 2D inverse Fourier transform operation.

If the input object is moved laterally in the input plane, the Fourier transform remains fixed in space but is multiplied by a phase factor that depends on the lateral movement. Therefore, the coordinates of the bright correlation output is proportional to the coordinates of the signal $f(\sigma, \varepsilon)$ located at the input plane. The intensity of the bright correlation spot is proportional to the degree to which the input and the filter functions are matched. This is also valid for multiple objects present at different locations of the input plane. This correlation system provides a great deal of sensitivity since it is both phase matched and amplitude matched [9–11].

For the complex matched filter (CMF), the frequency plane filter function is expressed by

$$H_{\text{cmf}} = R^*(o, v) = |R(o, v)| \exp[-j\phi(o, v)] \quad (3)$$

where, $|R(o, v)|$ is the amplitude and $\phi(o, v)$ is the phase factor of the Fourier spectrum of the reference function $r(\sigma, \varepsilon)$. When the input is similar to $r(\sigma, \varepsilon)$, the phase variation is canceled at the Fourier plane, thus producing a plane wave of light. The correlation peak corresponding to CMF is not very sharp due to the squaring of the magnitude of $r(\sigma, \varepsilon)$. Consequently, the resulting diffraction efficiency of a CMF is very poor. This filter is also unacceptably sensitive to even small changes in the reference signal or image. Currently available spatial light modulators (SLMs) cannot accommodate the full complex frequency response needed by CMFs.

3. Phase Only Filter (POF)

The optimum case with respect to light efficiency for a matched filter is realized by a phase only filter (POF) structure [10]. This filter is obtained by omitting the amplitude information and is defined as

$$H_{\text{pof}}(o, v) = \frac{R^*(o, v)}{|R(o, v)|} = \exp[-j\phi(o, v)] \quad (4)$$

Besides the improvement in light efficiency, the correlation peak intensity is also enhanced with a POF. However, although the autocorrelation peak intensity is higher than that of a CMF, it is not as sharp as might be produced by an IF since the product $H_{\text{pof}}(o, v)R(o, v)$ is not necessarily a constant.

4. Amplitude-Modulated Phase Only Filter (AMPOF)

Amplitude modulated phase only filter (AMPOF) [12] is given by

$$H_{\text{ampof}}(o, v) = \frac{A}{B + |R(o, v)|} \exp[-j\phi(o, v)] \quad (5)$$

where A and B are either constants or functions of o and v . The gain control factor A guarantees that the transmittance of the filter is less than unity. With the inclusion of B ; the pole problem is solved and at the same time it is possible to yield very high autocorrelation peak.

5. Synthetic Discriminant Functions (SDF)

The SDF-based correlation filters had shown robust performance for distortion tolerant pattern recognition [13–15]. Assume $x_1(\sigma, \varepsilon)$, $x_2(\sigma, \varepsilon)$, ..., $x_N(\sigma, \varepsilon)$ denote N training images representing possible distortions to a reference image $x(\sigma, \varepsilon)$. The 2D Fourier transform of $x(\sigma, \varepsilon)$ may be expressed as

$$X(o, v) = \iint x(\sigma, \varepsilon) \exp[-j2\pi(o\sigma + v\varepsilon)] \times d\sigma d\varepsilon \quad (6)$$

A composite image $h(\sigma, \varepsilon)$ is designed from the training images such that when the complex conjugate of its Fourier transform, denoted as $H^*(o, v)$, is correlated with input Fourier transform, a similar output is obtained for all N inputs, $x_1(\sigma, \varepsilon)$, $x_2(\sigma, \varepsilon)$, ..., $x_N(\sigma, \varepsilon)$. This type of correlator is known as frequency plane correlator. For this filter, the resulting spatial domain correlation output $c(\tau_\sigma, \tau_\varepsilon)$ may be expressed as [13]

$$\begin{aligned} c(\tau_\sigma, \tau_\varepsilon) &= \iint F(o, v) H^*(o, v) \exp[j2\pi(o\tau_\sigma + v\tau_\varepsilon)] d o d v \\ &= \iint h^*(\sigma, \varepsilon) f(\sigma + \tau_\sigma, \varepsilon + \tau_\varepsilon) d\sigma d\varepsilon \\ &= h(\sigma, \varepsilon) \odot f(\sigma, \varepsilon) \end{aligned} \quad (7)$$

where \odot denotes a two-dimensional cross-correlation operation.

In the Equal Correlation Peak SDF (ECP-SDF) design, the objective is to select a filter impulse response $h(\sigma, \varepsilon)$ such that the resulting crosscorrelations with all the N training images are the same, which is impossible in practice. Hester and Casasent [13] introduced a technique that requires that only the values at the origin of these crosscorrelations should be the same as shown in the following equation,

$$h(\sigma, \varepsilon) \odot x_i(\sigma, \varepsilon) \Big|_{\tau_\sigma=0, \tau_\varepsilon=0} = \iint h^*(\sigma, \varepsilon) x_i(\sigma, \varepsilon) d\sigma d\varepsilon = c, \quad i=1, 2, \dots, N \quad (8)$$

where c is a prespecified constant. Equation (8) shows that a $h(\sigma, \varepsilon)$ would yield the same constant value c at the origin (location of the autocorrelation peak) for all N training images (*i.e.*, $x_1(\sigma, \varepsilon)$ to $x_N(\sigma, \varepsilon)$). When the input is a non-training image from the same class, the crosscorrelation output at the origin will be similar to this constant c and it can be recognized. The success of this approach depends on selecting a proper training set.

Assume that $h(\sigma, \varepsilon)$ is a linear combination of the N training images, given by

$$h(\sigma, \varepsilon) = a_1 x_1(\sigma, \varepsilon) + \dots + a_N x_N(\sigma, \varepsilon) \quad (9)$$

where the coefficients a_1, a_2, \dots, a_N are determined in a way to satisfy the constraints of Equation (8). Substituting Equation (9) into Equation (8), we get

$$\sum_{i=1}^N a_i^* R_{ij} = c, \quad j=1, 2, \dots, N, \quad (10)$$

where

$$R_{ij} = \iint x_i^*(\sigma, \varepsilon) x_j(\sigma, \varepsilon) d\sigma d\varepsilon \quad (11)$$

is the inner product, *i.e.*, the crosscorrelation at the origin of the training images $x_i(\sigma, \varepsilon)$ and $x_j(\sigma, \varepsilon)$. If the training images are real, then there is no need for the conjugate operation shown in Equation (11). Equation (10) represents N complex linear equations with N complex unknowns, a_1, a_2, \dots, a_N , respectively. These equations can be solved by using any standard techniques, such as Gauss-Seidel Elimination method.

6. Modified Synthetic Discriminant Functions

The ECP-SDF is designed to produce a value c_i at the origin of the output plane when the i -th training image is used as the input. However, there are some practical problems in using this filter for pattern recognition applications [16–18]. Consequently, some improvements on ECP_SDF are suggested which are introduced in the following subsections.

6.1. Generalized SDF

Assume that the training images $x_1(\sigma, \varepsilon), x_2(\sigma, \varepsilon), \dots, x_N(\sigma, \varepsilon)$ are sampled to yield arrays $x_1(m, n), x_2(m, n), \dots, x_N(m, n)$ each with $\rho = \rho_1 \rho_2$ pixels, where ρ_1 is the number of pixels in the vertical direction while ρ_2 is the number of pixels in the horizontal direction of each image. It is also assumed that ρ -dimensional column vectors $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ are obtained by placing the elements in these training images in vectors where the scanning direction is from left to right and from top to bottom. Similarly, the ρ -dimensional column vector \mathbf{h} is used to denote the composite image $h(m, n)$. Then the constraints in Equation (8) can be rewritten as

$$\mathbf{h}^+ \mathbf{x}_i = c_i, \quad i=1, 2, 3, \dots, N \quad (12)$$

where the superscript $+$ denotes the conjugate transpose operation. The data matrix \mathbf{X} is assumed to have the vector \mathbf{x}_i as its i -th column and is thus a $\rho \times N$ matrix. It is also assumed that $\rho \gg N$, *i.e.*, the number of pixels in the training images is much larger than the number of training images, and that the columns of this matrix are linearly independent. Using this notation, Equation (12) can be rewritten as

$$\mathbf{X}^+ \mathbf{h} = \mathbf{c}^* \quad (13)$$

The ECP-SDF assumes that the composite image \mathbf{h} is of the following form

$$\mathbf{h} = \mathbf{X}\mathbf{a} \quad (14)$$

where \mathbf{a} is the vector of coefficients. Substituting Equation (14) into Equation (13) and solving for \mathbf{a} yields

$$\mathbf{a} = (\mathbf{X}^+ \mathbf{X})^{-1} \mathbf{c}^* \quad (15)$$

The filter vector \mathbf{h} can be obtained by substituting Equation (15) into Equation (14) to get

$$\mathbf{h}_{\text{ECP}} = \mathbf{X}(\mathbf{X}^+ \mathbf{X})^{-1} \mathbf{c}^* \quad (16)$$

A general expression for \mathbf{h} satisfying Equation (13) is given by

$$\mathbf{h}_{\text{GSDF}} = \mathbf{X}(\mathbf{X}^+ \mathbf{X})^{-1} \mathbf{c}^* + [\mathbf{I}_\rho - \mathbf{X}(\mathbf{X}^+ \mathbf{X})^{-1} \mathbf{X}^+] \mathbf{z} \quad (17)$$

where \mathbf{I}_ρ is the $\rho \times \rho$ diagonal identity matrix and \mathbf{z} is any column vector with ρ complex entries. The ECP-SDF is obtained from Equation (17) when $\mathbf{z} = 0$. The filter vector in Equation (17) is known as the generalized SDF [17].

6.2. Minimum Variance SDF

Consider a situation where the input image is one of the training images \mathbf{x}_i corrupted by additive noise \mathbf{n} . Then the resulting output value y , *i.e.*, the value of the crosscorrelation at the origin is given by

$$\begin{aligned} y &= \mathbf{h}^+ (\mathbf{x}_i + \mathbf{n}) \\ &= c_i + \mathbf{h}^+ \mathbf{n} \end{aligned} \quad (18)$$

where \mathbf{h} is designed to satisfy Equation (12). From Equation (18) it is evident that the output y is the desired output c_i corrupted by the random variable $(\mathbf{h}^+ \mathbf{n})$. The minimum variance synthetic discriminant function (MVSDF) [18] attempts to design \mathbf{h} such that the variance in the output caused by input noise is minimized while satisfying the constraints in Equation (13).

Assume that the real noise vector \mathbf{n} is a zero-mean vector with a $\rho \times \rho$ covariance matrix Σ . The variance of y corresponding to $\mathbf{h}^+ \mathbf{n}$ can be expressed as

$$\sigma_y^2 = E \left\{ |\mathbf{h}^+ \mathbf{n}|^2 \right\} = E \left\{ \mathbf{h}^+ \mathbf{n} \mathbf{n}^+ \mathbf{h} \right\} = \mathbf{h}^+ \Sigma \mathbf{h} \quad (19)$$

It is desired that σ_y^2 in Equation (19) is as small as possible, which will ensure that the output values are close to the constrained values even in the presence of noise. Minimizing σ_y^2 in Equation (19) subject to the constraints in Equation (13) leads to the following MVSDF [18]

$$\mathbf{h}_{\text{MVSDF}} = \Sigma^{-1} \mathbf{X} (\mathbf{X}^+ \Sigma^{-1} \mathbf{X})^{-1} \mathbf{c}^* \quad (20)$$

This MVSDF is indeed optimal from noise tolerance considerations. One difficulty in using this MVSDF is that often Σ is not known. Even when it is known, it is impossible to calculate its inversion. Another problem is that the MVSDF controls only one point (the origin) in the output-correlation plane. Thus, large sidelobes may be observed in the correlation output.

6.3. Frequency-Domain SDFs

It is often more convenient to design the filters in the frequency domain [16–18]. Assume $f(\sigma, \varepsilon)$ is the image and $H^*(o, \nu)$ is the complex filter function. Then the resulting correlation output $c(\tau_\sigma, \tau_\varepsilon)$ at the origin is given by

$$\begin{aligned} c(0,0) &= \iint H^*(o, \nu) F(o, \nu) d o d \nu \\ &= \hat{\mathbf{h}}^* \hat{\mathbf{f}} \end{aligned} \quad (21)$$

where $\hat{\cdot}$ indicates that the corresponding vector or matrix is obtained by sampling frequency domain functions and the superscript $^+$ indicates a conjugate transpose operation. Because $c_i(0,0)$ is constrained to be c_i , $i = 1, 2, \dots, N$, the constraints can be rewritten as

$$\hat{\mathbf{F}}^+ \hat{\mathbf{h}} = \mathbf{c}^* \quad (22)$$

where $\hat{\mathbf{F}}$ is a matrix with N columns with the i -th column containing $\hat{\mathbf{f}}_i$. It is obvious that the Equations (13) and (22) are similar.

6.4. Minimum Average Correlation Energy (MACE) Filter

The correlation filters discussed so far control only one point in the correlation plane. For good location accuracy and discrimination, it is necessary to design filters capable of producing sharp correlation peaks. One such filter is the minimum average correlation energy (MACE) filter [19]. Assume $x_1(\sigma, \varepsilon)$, $x_2(\sigma, \varepsilon)$, ..., $x_N(\sigma, \varepsilon)$ denote the N training images and $X_1(o, \nu)$, ..., $X_N(o, \nu)$ denote their 2D Fourier transforms, respectively. If $H^*(o, \nu)$ denotes the transmittance of the filter function, then the filter may be constructed to satisfy the following condition.

$$\iint X_i(o, \nu) H^*(o, \nu) d o d \nu = c_i, \quad i = 1, 2, \dots, N \quad (23)$$

In addition, the MACE filter minimizes the average correlation plane energy as shown below.

$$\begin{aligned} E_{\text{ave}} &= \frac{1}{N} \sum_{i=1}^N \iint |c_i(\tau_\sigma, \tau_\varepsilon)|^2 d \tau_\sigma d \tau_\varepsilon \\ &= \frac{1}{N} \sum_{i=1}^N \iint |X_i(o, \nu)|^2 |H(o, \nu)|^2 d o d \nu \end{aligned} \quad (24)$$

By minimizing E_{ave} , it is possible to keep the sidelobes in the correlation plane as low as possible. This is essentially an indirect attempt at reducing the problem of sidelobes. To carry out the minimization of E_{ave} , the usual vector notation is used. If $\hat{\mathbf{x}}_i$ denote the p -dimensional complex column vector obtained by sampling $X_i(o, \nu)$, then the constraints in Equation (23) can be rewritten as

$$\hat{\mathbf{X}}^+ \hat{\mathbf{h}} = \mathbf{c}^* \quad (25)$$

where $\hat{\mathbf{X}}$ is a $p \times N$ matrix with $\hat{\mathbf{x}}_i$ as its i -th column. The E_{ave} in Equation (24) can be expressed as

$$E_{\text{ave}} = \hat{\mathbf{h}}^+ \hat{\mathbf{D}} \hat{\mathbf{h}} \quad (26)$$

where $\hat{\mathbf{D}}$ is a $\rho \times \rho$ diagonal matrix. The entries along the diagonal are obtained by averaging $|X_i(o, v)|^2$, $i = 1, 2, \dots, N$, and then scanning the average from left to right and from top to bottom. Minimizing E_{ave} in Equation (26) subject to the constraints in Equation (25) leads to the following filter

$$\hat{\mathbf{h}}_{\text{MACE}} = \hat{\mathbf{D}}^{-1} \hat{\mathbf{X}} (\hat{\mathbf{X}}^+ \hat{\mathbf{D}}^{-1} \hat{\mathbf{X}})^{-1} \mathbf{c}^* \quad (27)$$

In many simulation studies, filters designed using this approach produced sharp correlation peaks. However, MACE filters appear to have two drawbacks. The first is that there is no noise tolerance built into these filters. The second is that these filters seem to be more sensitive to intra-class variations. Casasent *et al.* [20] proposed Gaussian MACE filters to reduce the sensitivity of the MACE filters to intra-class variations. The idea behind Gaussian MACE filters is to reduce the sharpness of the resulting correlation peak and thus improve its noise tolerance. MACE filters appear to be the first set of composite filters that attempt to control the entire correlation plane.

6.5. Minimum Squared Error SDF (MSE-SDF) Filter

This SDF design approach yields better approximation of arbitrary output correlation shapes in the minimum squared error (MSE) sense over the MACE filter and this filter is termed as MSE-SDF [21]. Like MACE, this filter must satisfy the usual SDF constraint of Equation (25). Besides, in MSE-SDF, the filter function $H(o, v)$ must make the correlation function c_i approximate a prespecified desired shape t_i , $i = 1, 2, \dots, N$. One measure of how well c_i approximates t_i is the average squared error E defined as

$$\begin{aligned} E &= \frac{1}{N} \sum_{i=1}^N \iint |t_i(\tau_\sigma, \tau_\epsilon) - c_i(\tau_\sigma, \tau_\epsilon)|^2 d\tau_\sigma d\tau_\epsilon \\ &= \frac{1}{N} \sum_{i=1}^N \iint |T_i(o, v) - X_i^*(o, v) H(o, v)|^2 do dv \end{aligned} \quad (28)$$

where T_i is the Fourier transform of t_i . If $\hat{\mathbf{X}}_i^{\mathbf{D}}$ is obtained by converting the vectors $\hat{\mathbf{x}}_i$ into diagonal matrices, then the average squared error of Equation (28) becomes

$$E = [E_d - \hat{\mathbf{h}}^+ \hat{\mathbf{p}} - \hat{\mathbf{p}}^+ \hat{\mathbf{h}} + \mathbf{h}^+ \hat{\mathbf{M}} \hat{\mathbf{h}}] \quad (29)$$

where

$$E_d = \frac{1}{N} \sum_{i=1}^N (\mathbf{t}_i^+ \mathbf{t}_i) \quad (30)$$

$$\hat{\mathbf{p}} = \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i^{\mathbf{D}} \mathbf{t}_i) \quad (31)$$

$$\hat{\mathbf{M}} = \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i^{\mathbf{D}*} \mathbf{X}_i^{\mathbf{D}}) \quad (32)$$

Minimizing E in Equation (29) subject to the constraints in Equation (25) leads to the following MSE-SDF filter

$$\hat{\mathbf{h}}_{\text{MSE-SDF}} = \hat{\mathbf{M}}^{-1} \hat{\mathbf{p}} + \hat{\mathbf{M}}^{-1} \hat{\mathbf{X}} \left[(\hat{\mathbf{F}}^+ \hat{\mathbf{M}}^{-1} \hat{\mathbf{F}})^{-1} \right] \left[\hat{\mathbf{c}} - \hat{\mathbf{F}}^+ \hat{\mathbf{M}}^{-1} \hat{\mathbf{p}} \right] \quad (33)$$

The MSE-SDF filter allows approximating arbitrary correlation shapes rather than zero shape implied in MACE filter design. This explicit control has two benefits [21]. First, correlation shapes can be selected with the linear/nonlinear postprocessing in mind and second, those correlation shapes that lead to better filter design can be used instead of simply minimizing the average correlation energy.

7. Maximum Average Correlation Height (MACH) Filter

The primary objective of the correlation filters is to achieve distortion-tolerant recognition of objects in the presence of clutter. This problem is easier to solve for in-plane rotations and scale changes. However, the prevalent method for handling out-of-plane distortions is to use a training set of representative views of the object. Traditionally, in the design of SDF-type correlation filters, linear constraints are imposed on the training images to yield a known value at specific locations in the correlation plane. However, placing such constraints in the correlation plane satisfies conditions only at isolated points in the image space but does not explicitly control the filter's ability to generalize over the entire domain of the training images. Various filters exhibit different levels of distortion tolerance even with the same training set and constraints.

The MACH filter adopts a statistical approach for filter design [22,23]. In addition to yielding sharp peaks and being computationally simple, this filter offers improved distortion tolerance. The reason lies in the fact that training images are not treated as deterministic representations of the object but as samples of a class whose characteristic parameters should be used in encoding the filter.

It is assumed that the training set consists of N images, and that each image of size $\rho_1 \times \rho_2$ contains $\rho = \rho_1 \rho_2$ pixels. The i -th training image for the target class is denoted by $x_i(m,n)$ in the spatial domain, which is represented in the frequency domain by a $\rho \times 1$ vector \mathbf{x}_i , obtained by lexicographically reordering its two-dimensional discrete Fourier transform, $X_i(k,l)$. The Fourier domain filter is denoted by the $\rho \times 1$ vector \mathbf{h} . The two-dimensional filter $H(k,l)$ is obtained by rearranging \mathbf{h} into a two-dimensional image. In this paper, matrices are denoted by uppercase bold-face and vectors by lowercase bold-face characters. The correlation of the i -th training image and the filter can be expressed in the frequency domain as

$$\mathbf{g}_i = \mathbf{X}_i \mathbf{h} \quad (34)$$

where \mathbf{X}_i is a $\rho \times \rho$ diagonal matrix containing the elements of \mathbf{x}_i . Here, \mathbf{g}_i denotes the discrete Fourier transform of the i -th correlation output. The deviation in the shape of the correlation plane with respect to some ideal shape vector \mathbf{f} is quantified by the average squared error (ASE), defined as

$$\text{ASE} = \frac{1}{N} \sum_{i=1}^N (\mathbf{g}_i - \mathbf{f})^+ (\mathbf{g}_i - \mathbf{f}) \quad (35)$$

Thus, ASE is a measure of distortion with respect to reference shape \mathbf{f} , which can be chosen as desired.

In fact, the shape vector \mathbf{f} can be treated as a free parameter in the distortion minimization problem. In the design of MSE-SDF [21], \mathbf{f} is specified as Gaussian or ring-like shapes in order to sculpt the correlation surface into these forms. In MACH, the choice of \mathbf{f} is such that it causes least variation

among the correlation planes and offers minimum ASE. To find the optimum shape \mathbf{f}_{opt} , the gradient of ASE with respect to \mathbf{f} is set to zero, given by

$$\nabla_{\mathbf{f}}(ASE) = \frac{2}{N} \sum_{i=1}^N (\mathbf{g}_i - \mathbf{f}) = 0 \quad (36)$$

or,

$$\mathbf{f}_{\text{opt}} = \frac{1}{N} \sum_{i=1}^N \mathbf{g}_i = \bar{\mathbf{g}} \quad (37)$$

where

$$\bar{\mathbf{g}} = \frac{1}{N} \sum_{i=1}^N \mathbf{X}_i \mathbf{h} = \mathbf{M} \mathbf{h} \quad (38)$$

is the average correlation plane and $\mathbf{M} = (1/N) \sum_{i=1}^N \mathbf{X}_i$ is the average training image expressed as a diagonal matrix. Thus, among all possible reference shapes, the average correlation plane $\bar{\mathbf{g}}$ offers the smallest possible ASE and the least distortion (in the squared error sense) among the correlation planes.

Substituting $\mathbf{f} = \bar{\mathbf{g}}$ in the ASE expression, the average similarity measure (ASM) is obtained as

$$\begin{aligned} \text{ASM} &= \frac{1}{N} \sum_{i=1}^N (\mathbf{g}_i - \bar{\mathbf{g}})^+ (\mathbf{g}_i - \bar{\mathbf{g}}) \\ &= \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i \mathbf{h} - \mathbf{M} \mathbf{h})^+ (\mathbf{X}_i \mathbf{h} - \mathbf{M} \mathbf{h}) \\ &= \mathbf{h}^+ \left[\sum_{i=1}^N (\mathbf{X}_i - \mathbf{M})^* (\mathbf{X}_i - \mathbf{M}) \right] \mathbf{h} \\ &= \mathbf{h}^+ \mathbf{S}_x \mathbf{h} \end{aligned} \quad (39)$$

where

$$\mathbf{S}_x = \sum_{i=1}^N (\mathbf{X}_i - \mathbf{M})^* (\mathbf{X}_i - \mathbf{M}) \quad (40)$$

is a diagonal matrix measuring the similarity of the training images to the class mean in the frequency domain. For example, if all training images are identical, then \mathbf{S}_x would be an all-zero matrix. From Parseval's theorem, it is easy to show that the average squared distance from the correlation planes to their mean is the same as that defined by Equation (39) in the frequency domain [22].

The ASM is one possible metric for distortion since it represents the average deviation of the correlation planes from the mean correlation shape, $\bar{\mathbf{g}}$. It is also a measure of the compactness of the class. If filter \mathbf{h} is viewed as a linear transform, then ASM measures the distances of the training images from the class center under this transform. Minimizing ASM, therefore, leads to a compact set of correlation planes that resemble each other and exhibit the least possible variations. The distortions of the object in the input plane are represented by the training images, \mathbf{x}_i . These distortions are reflected in the output as variations in the structure and shape of the corresponding correlation planes, \mathbf{g}_i , and are quantified by ASM. If the filter successfully reduces the distortions, then distorted input images should yield similar output planes, leading to a small value of ASM. Conversely, if ASM is

minimum and it is well shaped by design, then all true-class correlation planes are expected to resemble $\bar{\mathbf{g}}$ and to exhibit well-shaped structures.

The MACH filter relaxes the correlation peak constraints and maximizes the peak intensity of the average training image. The peak intensity of the average training image is $|\bar{\mathbf{g}}(0,0)|^2$ expressed as

$$|\bar{\mathbf{g}}(0,0)|^2 = \left| \frac{1}{N} \sum_{i=1}^N \mathbf{h}^+ \mathbf{x}_i \right|^2 = |\mathbf{h}^+ \mathbf{m}|^2 = \mathbf{h}^+ \mathbf{m} \mathbf{m}^+ \mathbf{h} \quad (41)$$

where \mathbf{m} is the Fourier transform of the average training image expressed as a vector.

Here, it is assumed without the loss of generality that the peak occurs at the origin of the correlation plane.

The smaller the value of ASM, the more invariant the response of the filter is. In other words, if ASM is small, then all true-class correlation planes are expected to resemble $\bar{\mathbf{g}}$. Therefore, it is required by \mathbf{h} to produce high correlation peak with the mean image while making ASM small. In addition, it is also required to obtain some degree of noise tolerance to reduce the output noise variance (ONV). For additive input noise, $\text{ONV} = \mathbf{h}^+ \mathbf{D} \mathbf{h}$, where \mathbf{D} is the diagonal power spectral density matrix [23]. While practical noise may be multiplicative and more complicated than implied by simple additive noise, the simple additive noise model at least provides some robustness. The performance criterion used to optimize the MACH filter may be expressed as

$$\begin{aligned} J(\mathbf{h}) &= \frac{(\text{Average peak height})^2}{\text{ASM} + \text{ONV}} \\ &= \frac{|\bar{\mathbf{g}}(0,0)|^2}{\text{ASM} + \text{ONV}} \\ &= \frac{|\mathbf{h}^+ \mathbf{m}|^2}{\mathbf{h}^+ \mathbf{S} \mathbf{h} + \mathbf{h}^+ \mathbf{D} \mathbf{h}} \\ &= \frac{\mathbf{h}^+ \mathbf{m} \mathbf{m}^+ \mathbf{h}}{\mathbf{h}^+ (\mathbf{S} + \mathbf{D}) \mathbf{h}} \end{aligned} \quad (42)$$

The optimum solution is found by setting the derivative of $J(\mathbf{h})$ in Equation (42) with respect to \mathbf{h} to zero and is given by [22,23]

$$\mathbf{h} = (\mathbf{S} + \mathbf{D})^{-1} \mathbf{m} \quad (43)$$

The filter in Equation (43) is referred to as the MACH filter because it maximizes the height of the mean correlation peak relative to the expected distortions. For cases where an estimate of \mathbf{D} is not available, the white noise covariance matrix is substituted for \mathbf{D} , *i.e.*, $\mathbf{D} = \sigma^2 \mathbf{I}$, where \mathbf{I} is a diagonal identity matrix. Hence the simplified MACH filter becomes

$$\mathbf{h} = (\mathbf{S} + \sigma^2 \mathbf{I})^{-1} \mathbf{m} \quad (44)$$

Replacing σ^2 by another constant γ , the filter equation becomes

$$\mathbf{h} = (\mathbf{S} + \gamma \mathbf{I})^{-1} \mathbf{m} \quad (45)$$

The robustness of the MACH filter is attributed to the inclusion of the ASM criterion, which reduces the filter sensitivity to distortions, and to the removal of hard constraints on the peak. The later

fact enables the correlation planes to adjust to suitable values for optimizing the performance criterion. However, MACH filter can handle those distortions that are well represented in the training set.

The Fourier domain MACH filter obtained in Equation (45) may be converted to the 2D shape or size of the input training images expressed as $H(k,l)$. A 2D test image $z(m,n)$ is Fourier transformed to obtain $Z(k,l)$ which is then correlated with the 2D filter in the Fourier domain using the expression

$$G(k,l) = Z(k,l)H^*(k,l) \quad (46)$$

The output spatial domain correlation is obtained by applying the inverse Fourier transform operation to Equation (46) and recording the intensity, given by

$$g(m,n) = |\mathcal{F}^{-1}[G(k,l)]|^2 \quad (47)$$

8. Extended MACH (EMACH) Filter

The average training image used in the MACH filter design is good in representing the average behavior of the desired class, but it fails to capture the finer details of the desired class [24,25]. In fact, the average of training images sometimes looks like a clutter image. Thus, the MACH filter may be inadequate in discriminating the desired class from the clutter, leading to increased false alarm rate. The extended MACH (EMACH) filter is aimed at improving this clutter rejection capability.

The MACH filter is designed to maximize the intensity of the average correlation output at the origin due to training images. The average of correlation peaks is the correlation output due to the average training image. It also maximizes the similarity between the average training image correlation output and those outputs due to all training images from the desired class. Thus, the MACH filter forces all images from the desired class to follow the behavior of the average training image from that class. The MACH filter relies heavily on the mean training image. It amplifies the high-energy (usually low-frequency) components, and at the same time, attenuates the low-energy (usually high-frequency) components of the training set. Thus, by using the mean image \mathbf{m} as the only example that represents all training images, a filter may be obtained that does not capture the finer details of the training images. Therefore, this filter may fail to discriminate the desired class from the clutter. The MACH filters possess attributes that may lead to detect clutter images as targets. One such attribute is that all training images follow the same behavior as the average training image. However, the average training image is not necessarily a good representative of the desired class.

To control the relative contribution of the desired class training images as well as their average, a new metric, called all image correlation height (AICH) [25], is introduced and defined as

$$\begin{aligned} \text{AICH} &= \frac{1}{N} \sum_{i=1}^N [\mathbf{h}^+ (\mathbf{x}_i - \beta \mathbf{m})]^{-2} \\ &= \frac{1}{N} \sum_{i=1}^N (\mathbf{h}^+ \mathbf{x}_i - \beta \mathbf{h}^+ \mathbf{m})^2 \end{aligned} \quad (48)$$

where β is a parameter that takes a value between 0 and 1 and governs the relative significance of the average training image in the filter design. By controlling β , the designed filter is prevented from being overwhelmed by the biased treatment of the low-frequency components represented by the average image. Here, the AICH must be optimized and to be able to do that, Equation (48) may be rewritten as

$$\begin{aligned}
\text{AICH} &= \frac{1}{N} \sum_{i=1}^N (\mathbf{h}^+ \mathbf{x}_i - \beta \mathbf{h}^+ \mathbf{m}) (\mathbf{h}^+ \mathbf{x}_i - \beta \mathbf{h}^+ \mathbf{m})^+ \\
&= \mathbf{h}^+ \left[\frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \beta \mathbf{m}) (\mathbf{x}_i - \beta \mathbf{m})^+ \right] \mathbf{h} \\
&= \mathbf{h}^+ \mathbf{C}_x^\beta \mathbf{h}
\end{aligned} \tag{49}$$

where

$$\mathbf{C}_x^\beta = \frac{1}{N} \sum_{i=1}^N (\mathbf{x}_i - \beta \mathbf{m}) (\mathbf{x}_i - \beta \mathbf{m})^+ \tag{50}$$

Thus, AICH can be described as the average of the correlation peak intensities of N exemplars where the i -th exemplar $(\mathbf{x}_i - \beta \mathbf{m})$ is the i -th training image with part of the mean subtracted. Hence, it is desirable for all images in the training set to follow these exemplars' behavior. This can be done by forcing every image in the training set \mathbf{x}_i to have a similar correlation output plane to an ideal correlation output shape \mathbf{f} . To find the \mathbf{f} that best matches all these exemplars' correlation output planes, its deviation from their correlation planes is minimized. This deviation can be quantified by ASE, defined as

$$\text{ASE} = \frac{1}{N} \sum_{i=1}^N (\mathbf{g}_i - \mathbf{f})^+ (\mathbf{g}_i - \mathbf{f}) \tag{51}$$

where,

$$\mathbf{g}_i = (\mathbf{X}_i - \beta \mathbf{M}) \mathbf{h}^* \tag{52}$$

In Equation (52), the superscript $*$ represents the complex conjugate operation. To find the optimum shape vector \mathbf{f}_{opt} , the gradient of ASE with respect to \mathbf{f} is set to zero, yielding

$$\begin{aligned}
\mathbf{f}_{\text{opt}} &= \frac{1}{N} \mathbf{g}_i \\
&= \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i - \beta \mathbf{M}) \mathbf{h}^* \\
&= (1 - \beta) \mathbf{M} \mathbf{h}^*
\end{aligned} \tag{53}$$

The ASM is modified such that it measures the dissimilarity of the training images to $(1 - \beta) \mathbf{M} \mathbf{h}^*$. This new measure is called as the modified ASM (MASM), given by

$$\begin{aligned}
\text{MASM} &= \frac{1}{N} \sum_{i=1}^N [\mathbf{X}_i \mathbf{h}^* - (1 - \beta) \mathbf{M} \mathbf{h}^*]^+ [\mathbf{X}_i \mathbf{h}^* - (1 - \beta) \mathbf{M} \mathbf{h}^*] \\
&= \mathbf{h}' \left\{ \frac{1}{N} \sum_{i=1}^N [\mathbf{X}_i - (1 - \beta) \mathbf{M}]^* [\mathbf{X}_i - (1 - \beta) \mathbf{M}] \right\} \mathbf{h}^* \\
&= \mathbf{h}' \mathbf{S}_x^\beta \mathbf{h}^* \\
&= \mathbf{h}^+ \mathbf{S}_x^\beta \mathbf{h}
\end{aligned} \tag{54}$$

where the superscript $'$ represents the transpose operation and where it is considered that MASM is real in deriving the last equality in Equation (54). The diagonal matrix \mathbf{S}_x^β is given by

$$\mathbf{S}_x^\beta = \frac{1}{N} \sum_{i=1}^N [\mathbf{X}_i - (1-\beta)\mathbf{M}]^* [\mathbf{X}_i - (1-\beta)\mathbf{M}] \quad (55)$$

The ASM is a good measure for distortion tolerance; however, it lacks some discrimination capability that explains part of the MACH filter's inability to reject some clutter images. On the other hand, the MASM measure captures finer details of the training set that makes the EMACH filter more sensitive against clutter.

By maximizing the AICH and minimizing the MASM while controlling the parameter β , it is expected to explicitly keep a balance between the distortion tolerance and clutter rejection performance. Therefore, it is necessary to optimize the following new criterion

$$J^\beta(\mathbf{h}) = \frac{\text{AICH}}{\mathbf{h}^+ \boldsymbol{\gamma} \mathbf{I} \mathbf{h} + \mathbf{h}^+ \mathbf{S}_x^\beta \mathbf{h}} = \frac{\mathbf{h}^+ \mathbf{C}_x^\beta \mathbf{h}}{\mathbf{h}^+ (\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta) \mathbf{h}} \quad (56)$$

where $\mathbf{h}^+ \boldsymbol{\gamma} \mathbf{I} \mathbf{h}$ is the ONV term assuming an additive white noise with variance γ . The ONV helps to maintain noise tolerance when β increases, especially, at those low energy components. By maximizing the preceding criterion, the following condition is obtained for the EMACH filter

$$(\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta)^{-1} \mathbf{C}_x^\beta \mathbf{h} = \lambda \mathbf{h} \quad (57)$$

where λ is a scalar identical to $J^\beta(\mathbf{h})$. Thus, \mathbf{h} must be an eigenvector of $(\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta)^{-1} \mathbf{C}_x^\beta$ with the corresponding eigenvalue λ . Since λ is identical to $J^\beta(\mathbf{h})$, \mathbf{h} should be the eigenvector that corresponds to the maximum eigenvalue. The other eigenvectors corresponding to the other nonzero eigenvalues provide smaller $J^\beta(\mathbf{h})$ values. However, they may provide better discriminatory performance, as β is not known *a priori*. So the EMACH filter may be expressed as

$$\mathbf{h} = \text{Dominant eigenvector} \{ (\boldsymbol{\gamma} \mathbf{I} + \mathbf{S}_x^\beta)^{-1} \mathbf{C}_x^\beta \} \quad (58)$$

9. Distance Classifier Correlation Filter (DCCF)

This is a correlation-based distance classifier scheme for recognition and classification of multiple similar or dissimilar objects. The underlying theory uses shift-invariant filters to compute distances between the input image and ideal references under an optimum transformation. The two ideas of relaxing the constraints on the correlation values at the origin and looking at the entire correlation plane rather than just the peak value led to the development of distance classifier correlation filters DCCFs [26–28]. The DCCF formulation can be used with any number of classes.

In the DCCF design, a global transformation is determined such that the transformed images from the same class are close to each other, whereas transformed images from different classes are separated from each other. This global transform leads to one correlation filter for each class. The use of these correlation filters is similar to the use of other correlation filters except in the final step. The test image is correlated with the correlation filter, and the resulting correlation peak is determined. This correlation-peak value is used to determine the distance of the test image to this class. Distances of the test image to all classes are determined and the class yielding the smallest distance is chosen. This paradigm allows for the relaxation of the correlation output constraints and the use of the entire correlation output.

It is important to realize that the use of DCCFs is similar to the use of other correlation filters. This means that the correlation peaks move by the same amount corresponding to the shift in the input, *i.e.*, this is a shift-invariant operation. The DCCF concept has demonstrated promising performance on both infrared as well as synthetic aperture radar imagery [28]. Test results show that DCCFs outperform other correlation filters in recognizing targets while rejecting noise, clutter, and other confusing objects.

It is assumed that the training images are segmented and registered at a desired point. Fourier transform of an image $x(m,n)$ of size $\rho_1 \times \rho_2$ containing $\rho = \rho_1\rho_2$ pixels can be expressed as a $\rho \times 1$ dimensional column vector \mathbf{x} or as a $\rho \times \rho$ diagonal matrix \mathbf{X} with the elements of \mathbf{x} as its diagonal elements, *i.e.*, diagonal $\{\mathbf{x}\} = \mathbf{X}$. Sometimes, the same quantity may be expressed both as a vector, say \mathbf{m}_x , and as a diagonal matrix \mathbf{M}_x . This implies that $\mathbf{H}\mathbf{m}_x$ and $\mathbf{M}_x\mathbf{h}$ are equivalent.

The distance classifier uses a global transform denoted by \mathbf{H} to separate the classes maximally while making them as compact as possible. For shift invariance, this transform matrix must be diagonal in the frequency domain. Multiplication of a vector \mathbf{x} by a diagonal matrix \mathbf{H} is equivalent to multiplying $X(k,l)$ by $H(k,l)$.

Here, a general C -class distance-classifier problem is analyzed by assuming that the peak correlation values are as different as possible for each of the classes although hard constraints are not used to enforce this. In addition, for each class, the correlation planes or their inverse Fourier transforms should be almost similar to the transformed ideal reference shape for this class. The correlation peaks are most likely at the origin for the registered training images but can occur elsewhere in the test cases depending on the location of the target.

Assume \mathbf{x}_{ik} is the ρ -dimensional column vector containing the Fourier transform of the i -th image of the k -th class, $1 \leq i \leq N$ and $1 \leq k \leq C$, where each class contains N training images. If \mathbf{m}_k is the mean Fourier transform of class k , then

$$\mathbf{m}_k = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_{ik}, \quad 1 \leq k \leq C \quad (59)$$

The correlation peak at the origin between the mean training image from the k -th class and the filter \mathbf{h} is given by $\mathbf{m}_k^+\mathbf{h}$. Thus, the overall mean Fourier transform of the entire training set becomes

$$\mathbf{m} = \frac{1}{C} \sum_{k=1}^C \mathbf{m}_k \quad (60)$$

The correlation peak of the overall mean image \mathbf{m} with the filter \mathbf{h} is given by

$$\mathbf{m}^+\mathbf{h} = \frac{1}{C} \sum_{k=1}^C \mathbf{m}_k^+\mathbf{h} \quad (61)$$

which represents the overall average of the origin values of all correlation planes.

If the transformation \mathbf{h} makes the in-class correlation planes similar, then the in-class peak values should be similar to each other and to their mean. Thus, to make the interclass separation between the correlation peaks large, the mean peak values of the classes are made as different as possible. Although several possible criteria might achieve this objective, the approach here is to increase the distance of all classes from the central mean. Toward this end, the following distance measure, called class separation, has been formulated

$$\begin{aligned}
 A(\mathbf{h}) &= \frac{1}{C} \sum_{k=1}^C |\mathbf{m}_k^+ \mathbf{h} - \mathbf{m}^+ \mathbf{h}|^2 \\
 &= \frac{1}{C} \sum_{k=1}^C \mathbf{h}^+ (\mathbf{m} - \mathbf{m}_k) (\mathbf{m} - \mathbf{m}_k)^+ \mathbf{h} \\
 &= \mathbf{h}^+ \mathbf{W} \mathbf{h}
 \end{aligned} \tag{62}$$

where

$$\mathbf{W} = \frac{1}{C} \sum_{k=1}^C (\mathbf{m} - \mathbf{m}_k) (\mathbf{m} - \mathbf{m}_k)^+ \tag{63}$$

is a $\rho \times \rho$, full (*i.e.*, non-diagonal) matrix of rank less than or equal to $(C-1)$. The rank of \mathbf{W} is less than or equal to $(C-1)$, because it is obtained by the addition of C outer products of vectors $(\mathbf{m} - \mathbf{m}_k)$, but these C vectors add up to a zero vector. If $A(\mathbf{h})$ of Equation (62) is maximized, the class mean correlation peaks $(\mathbf{m}_k^+ \mathbf{h})$ will differ significantly. It is also desired that the distance of transformed inputs to their average be small. This distance $B(\mathbf{h})$, which measures the compactness of each class, is the same as ASM defined for each class as

$$\text{ASM}_k = \frac{1}{N} \sum_{i=1}^N |\mathbf{g}_{ik} - \bar{\mathbf{g}}_k|^2 \quad 1 \leq k \leq C \tag{64}$$

where

$$\begin{aligned}
 \mathbf{g}_{ik} &= \mathbf{X}_{ik} \mathbf{h}^* \\
 \bar{\mathbf{g}}_k &= \mathbf{M}_k \mathbf{h}^*
 \end{aligned} \tag{65}$$

are the Fourier transforms of the correlation outputs due to the i -th training image \mathbf{x}_{ik} and the average training image \mathbf{m}_k , respectively from class k . Note that \mathbf{X}_{ik} and \mathbf{M}_k are diagonal matrices with \mathbf{x}_{ik} and \mathbf{m}_k along the diagonal. The ASM is a measure of the similarity of the training images of a class to their mean and hence a measure of the compactness of the class after transformation by \mathbf{H} . Using Equations (64) and (65), ASM_k can be rewritten as

$$\begin{aligned}
 \text{ASM}_k &= \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_{ik} \mathbf{h}^* - \mathbf{M}_k \mathbf{h}^*)^+ (\mathbf{X}_{ik} \mathbf{h}^* - \mathbf{M}_k \mathbf{h}^*) \\
 &= \mathbf{h}' \left[\frac{1}{N} \sum_{i=1}^N (\mathbf{X}_{ik} - \mathbf{M}_k)^* (\mathbf{X}_{ik} - \mathbf{M}_k) \right] \mathbf{h}^* \\
 &= \mathbf{h}^+ \mathbf{S}_k \mathbf{h}
 \end{aligned} \tag{66}$$

where

$$\mathbf{S}_k = \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_{ik} - \mathbf{M}_k)^* (\mathbf{X}_{ik} - \mathbf{M}_k) \tag{67}$$

In Equation (67), \mathbf{S}_k is a $\rho \times \rho$ diagonal matrix where each training image contains ρ pixels. The overall ASM for C classes is defined as

$$\text{ASM} = B(\mathbf{h}) = \frac{1}{N} \sum_{k=1}^C \mathbf{h}^+ \mathbf{S}_k \mathbf{h} = \mathbf{h}^+ \mathbf{S} \mathbf{h} \tag{68}$$

where

$$\mathbf{S} = \frac{1}{C} \sum_{k=1}^C \mathbf{S}_k \quad (69)$$

To make the in-class metric $B(\mathbf{h})$ small and to make the inter-class distance metric $A(\mathbf{h})$ large, the filter \mathbf{h} is designed to maximize the ratio

$$J(\mathbf{h}) = \frac{A(\mathbf{h})}{B(\mathbf{h})} = \frac{\mathbf{h}^+ \mathbf{W} \mathbf{h}}{\mathbf{h}^+ \mathbf{S} \mathbf{h}} \quad (70)$$

with respect to \mathbf{h} . The filter \mathbf{h} that maximizes $J(\mathbf{h})$ in Equation (70) is the eigenvector of $\mathbf{S}^{-1} \mathbf{W}$ with the largest eigenvalue [28]. Because \mathbf{W} is a non-diagonal matrix of rank less than or equal to $(C-1)$, finding the dominant eigenvector of $\mathbf{S}^{-1} \mathbf{W}$ requires a special algorithm when the training images, and thus the desired filter is of larger size [28]. When $J(\mathbf{h})$ is maximum, the correlation shape produced by an input image is expected to be similar to the mean shape for its true class with a peak value different from the average peak value of any other class. The DCCF filter may be expressed as

$$\mathbf{h} = \text{Dominant eigenvector}\{\mathbf{S}^{-1} \mathbf{W}\} \quad (71)$$

The DCCF is the first technique proposed for shift-invariant transform-domain distance calculations with a correlator and that are specifically designed to accommodate multiple targets at different locations in the same image. This filter deals with the entire correlation plane and not just one point at the origin. It transforms the input image into a new space in which the distance of a test input from the classes is computed. Given a test input \mathbf{z} , the distances d_k between the transformed input and the ideal shape for class k is computed as

$$\begin{aligned} d_k &= \left| \mathbf{H}^* \mathbf{z} - \mathbf{H}^* \mathbf{m}_k \right|^2 \\ &= \left| \mathbf{H}^* \mathbf{z} \right|^2 + \left| \mathbf{H}^* \mathbf{m}_k \right|^2 - 2\Re\{\mathbf{z}^+ \mathbf{H} \mathbf{H}^* \mathbf{m}_k\} \\ &= p + b_k - 2\Re\{\mathbf{z}^+ \mathbf{h}_k\} \end{aligned} \quad (72)$$

In Equation (72), $p = \left| \mathbf{H}^* \mathbf{z} \right|^2$ is the energy (independent of the class) of the transformed input test image \mathbf{z} ; $b_k = \left| \mathbf{H}^* \mathbf{m}_k \right|^2$ is the energy (independent of the \mathbf{z}) of the transformed mean of class k and $\mathbf{h}_k = \mathbf{H} \mathbf{H}^* \mathbf{m}_k$ is viewed as the effective filter for class k . Because there are only C classes for which distances must be computed, only C such filters are required.

In general, the targets may be anywhere in the input image. For shift-invariant distance calculation the interest is in the smallest value of d_k over all possible shifts of the target with respect to the class references. In Equation (72), because p and b_k are both positive and independent of the position of the target, the smallest value of d_k over all shifts is obtained when the third term (*i.e.*, $\mathbf{z}^+ \mathbf{h}_k$) is as large as possible. Therefore, this term is chosen as the peak value for full cross correlation of \mathbf{z} and \mathbf{h}_k .

10. Polynomial DCCF (PDCCF)

Linear transformations such as the DCCF are attractive because of their optimality when the underlying statistics are Gaussian with equal covariances [29]. However, the DCCF uses transformations based on

the second order statistics and does not capture higher-order statistics in images. Hence, it does not necessarily capture all of the discrimination information in some cases. Examples of such cases are encountered when signal dependent or multiplicative noises are present, and when inputs have non-Gaussian statistics. In non-Gaussian statistics cases, DCCF capabilities may be improved by applying different nonlinearities to the input image. By using nonlinear transformations, it may be possible to extract more useful information for discrimination. Thus, the classes that are not well separated in the original image space may become more separated in the nonlinearly mapped space. Also, point nonlinearities are preferred because they reduce the computational complexity because a simple nonlinearity is being applied to each point without considering all points in the neighborhood.

The polynomial DCCF (PDCCF) extends the DCCF to include point nonlinear mappings of the input patterns [29]. Examples of such nonlinear mappings correspond to powers of pixels of the input images. Even though the resulting PDCCF system is not linear with respect to the input patterns, it is still linear in the kernel. This property allows frequency domain techniques for the design, analysis, and implementation of this filter. Another important property is that this system works on different powers of input image pixels, which corresponds to a multi-dimensional correlation operation and thus extends the linear DCCF classification optimization criterion to a nonlinear one, and no nonlinear optimization is involved. Moreover, the PDCCF system provides a new framework for combining different correlation filters, where each filter in the system is optimized jointly with other filters.

The PDCCF first maps the input image $x(m,n)$ into $x^j(m,n)$ via point nonlinearities η_j , where $j = 1, 2, \dots, n$. Thus $x^j(m,n)$ is related to $x(m,n)$ through the following relationship

$$\eta_j : x(m,n) \rightarrow x^j(m,n) \quad (73)$$

In Equation (73), all $x^j(m,n)$ are assumed to have the same size as the input $x(m,n)$. Examples of such mappings include various powers, logarithms, cosines, *etc.* The nonlinearly mapped input image $x(m,n)$ is transformed by the filter $h_j(m,n)$ built using $x^j(m,n)$. The overall distance is obtained by adding the distances resulting from the shift-invariant minimum mean squared error computations between every transformed image of the input and its respective ideal transformed reference. Those ideal transformed references are computed *a priori* by using the nonlinear functions η_1, η_2, \dots , and η_n followed by the application of the filters $h_1(m,n), h_2(m,n), \dots$, and $h_n(m,n)$, respectively.

In the linear DCCF, the filtered image $g(m,n)$ resulting from transformation of $x^1(m,n)$ by $h_1(m,n)$ can be written as

$$g(m,n) = h_1(m,n) \odot x^1(m,n) \quad (74)$$

where, \odot denotes the crosscorrelation and $x^1(m,n) = x(m,n)$. By augmenting the input image with $x^2(m,n)$, another term can be added involving the crosscorrelation of the $x^2(m,n)$ with $h_2(m,n)$ to obtain an output transformed by the two-term PDCCF, as shown below.

$$g(m,n) = h_1(m,n) \odot x^1(m,n) + h_2(m,n) \odot x^2(m,n) \quad (75)$$

Thus, the PDCCF has more terms at its disposal with which it can achieve better discrimination. Clearly, these new nonlinear versions of inputs are completely dependent on the original inputs, and in that sense no new information is being created. However, the new representations enable the correlation filters to provide better recognition. By continuing to add more terms such as $h_j(m,n) \odot x^j(m,n)$, the n -term PDCCF can be obtained as

$$g(m, n) = \sum_{j=1}^n h_j(m, n) \odot x^j(m, n) \quad (76)$$

If the focus is on the point-wise power nonlinearities for all η_j 's the nonlinear mapping of the input, $x^j(m, n)$ can be defined as

$$x^j(m, n) = [x(m, n)]^j \quad (77)$$

where, $j \in \mathbb{R}^{\infty}$. The power nonlinearity plays an important role in SAR images as it can enhance the bright scatters or the overall contrast [29]. The filters $h_j(m, n)$ are computed jointly, and thus the advantages of the closed form solutions and nonlinear systems are combined and exploited. In the following analysis, Ψ represents the set of all powers used to construct a particular PDCCF. Although the nonlinearity is applied in the spatial domain, in the analyses to follow (e.g., filter formation, distance calculation, *etc.*) all the quantities are actually in Fourier domain. For example, \mathbf{x} is a vector obtained by lexicographical rearrangement of the Fourier transform of $x(m, n)$. All vectors and matrices with superscript j (e.g., \mathbf{x}^j and \mathbf{X}^j) represent these vectors and matrices (\mathbf{x} and \mathbf{X}) with each of their elements in the image (spatial) domain raised to the j -th power.

Assume \mathbf{m}_k^j is the mean of class k of the Fourier transforms of training images resulting from raising all their pixels to the j -th power and \mathbf{h}_j is the filter built for images raised to the j -th power. Each of the Fourier transforms of the original image along with the Fourier transforms of its variations can be represented by use of a single block vector. Thus, the Fourier transform of the mean images of class k after augmentation becomes

$$\mathbf{m}_k = \begin{pmatrix} \mathbf{m}_k^1 \\ \mathbf{m}_k^2 \\ \cdot \\ \cdot \\ \mathbf{m}_k^n \end{pmatrix} \quad (78)$$

Further, the filters, $\mathbf{h}_1, \mathbf{h}_2, \dots$ and \mathbf{h}_n can be combined into one filter, \mathbf{h} as follows

$$\mathbf{h} = \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \cdot \\ \cdot \\ \mathbf{h}_n \end{pmatrix} \quad (79)$$

With Equations (75), (78), and (79), the correlation peak at the origin produced in response to the mean image of class k is given by

$$\bar{g}_k(0, 0) = \sum_{j=1}^n \mathbf{h}_j^+ \mathbf{m}_k^j = \mathbf{h}^+ \mathbf{m}_k \quad (80)$$

The distance between the classes, after being augmented and then transformed by the filters $\mathbf{h}_1, \mathbf{h}_2, \dots$, and \mathbf{h}_n can be expressed as

$$\begin{aligned}
 A(\mathbf{h}) &= \frac{1}{C} \sum_{k=1}^C \left| \sum_{j=1}^n \mathbf{h}_j^+ \mathbf{m}_k^j - \sum_{j=1}^n \mathbf{h}_j^+ \mathbf{m}^j \right|^2 \\
 &= \frac{1}{C} \sum_{k=1}^C \left| \mathbf{h}^+ \mathbf{m}_k - \mathbf{h}^+ \mathbf{m} \right|^2 \\
 &= \mathbf{h}^+ \mathbf{W} \mathbf{h}
 \end{aligned} \tag{81}$$

where

$$\mathbf{W} = \frac{1}{C} \sum_{k=1}^C (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^+ \tag{82}$$

To separate the classes as much as possible, the filter is required to produce a large $A(\mathbf{h})$. Simultaneously, the compactness of the classes needs to be increased after transformation by $\mathbf{h}_1, \mathbf{h}_2, \dots$, and \mathbf{h}_n . The compactness is measured by the similarity of the training images of a class to their mean. It can be represented by the ASM of that class. In general, the ASM for class k is defined as

$$\text{ASM}_k = \frac{1}{N} \sum_{i=1}^N \left| \mathbf{g}_{ik} - \bar{\mathbf{g}}_k \right|^2 \quad 1 \leq k \leq C \tag{83}$$

where

$$\begin{aligned}
 \mathbf{g}_{ik} &= \mathbf{X}_{ik}^j \mathbf{h}_j^* \\
 \bar{\mathbf{g}}_k &= \mathbf{M}_k^j \mathbf{h}_j^*
 \end{aligned} \tag{84}$$

are the Fourier transforms of the filtered images produced by the transform filters in response to the input image \mathbf{x}^j and the mean image \mathbf{m}_k , respectively. Thus, from Equations (83) and (84), ASM_k can be written as

$$\begin{aligned}
 \text{ASM}_k &= \frac{1}{N} \sum_{i=1}^N \left| \mathbf{X}_{ik}^1 \mathbf{h}_1^* + \mathbf{X}_{ik}^2 \mathbf{h}_2^* + \dots + \mathbf{X}_{ik}^n \mathbf{h}_n^* - \mathbf{M}_k^1 \mathbf{h}_1^* - \mathbf{M}_k^2 \mathbf{h}_2^* - \dots - \mathbf{M}_k^n \mathbf{h}_n^* \right|^2 \\
 &= \frac{1}{N} \sum_{i=1}^N \left[(\mathbf{X}_{ik}^1 - \mathbf{M}_k^1) \mathbf{h}_1^* + (\mathbf{X}_{ik}^2 - \mathbf{M}_k^2) \mathbf{h}_2^* + \dots + (\mathbf{X}_{ik}^n - \mathbf{M}_k^n) \mathbf{h}_n^* \right]^+ \\
 &\quad \left[(\mathbf{X}_{ik}^1 - \mathbf{M}_k^1) \mathbf{h}_1^* + (\mathbf{X}_{ik}^2 - \mathbf{M}_k^2) \mathbf{h}_2^* + \dots + (\mathbf{X}_{ik}^n - \mathbf{M}_k^n) \mathbf{h}_n^* \right] \\
 &= \sum_{u=1}^n \sum_{v=1}^n \mathbf{h}_u^+ \mathbf{S}_{kuv} \mathbf{h}_v \\
 &= (\mathbf{h}_1^+ \mathbf{h}_2^+ \dots \mathbf{h}_n^+) \cdot \begin{bmatrix} \mathbf{S}_{k11} & \mathbf{S}_{k12} & \dots & \mathbf{S}_{k1n} \\ \mathbf{S}_{k21} & \mathbf{S}_{k22} & \dots & \mathbf{S}_{k2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{kn1} & \mathbf{S}_{kn2} & \dots & \mathbf{S}_{knn} \end{bmatrix} \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_n \end{pmatrix} \\
 &= \mathbf{h}^+ \mathbf{S}_k \mathbf{h}
 \end{aligned} \tag{85}$$

where

$$\mathbf{S}_{kuv} = \left[\frac{1}{N} \sum_{i=1}^N \mathbf{X}_{ik}^u (\mathbf{X}_{ik}^v)^* \right] - \mathbf{M}_k^u (\mathbf{M}_k^v)^* \quad (86)$$

are all diagonal matrices and

$$\mathbf{S}_k = \begin{bmatrix} \mathbf{S}_{k11} & \mathbf{S}_{k12} & \cdots & \mathbf{S}_{k1n} \\ \mathbf{S}_{k21} & \mathbf{S}_{k22} & \cdots & \mathbf{S}_{k2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{kn1} & \mathbf{S}_{kn2} & \cdots & \mathbf{S}_{knn} \end{bmatrix} \quad (87)$$

The overall ASM for C classes is then defined as

$$\text{ASM} = B(\mathbf{h}) = \frac{1}{N} \sum_{k=1}^C \mathbf{h}^+ \mathbf{S}_k \mathbf{h} = \mathbf{h}^+ \mathbf{S} \mathbf{h} \quad (88)$$

where

$$\mathbf{S} = \frac{1}{C} \sum_{k=1}^C \mathbf{S}_k \quad (89)$$

The filter \mathbf{h} that maximizes the ratio $A(\mathbf{h})/B(\mathbf{h})$, is the dominant eigenvector of $\mathbf{S}^{-1}\mathbf{W}$ as in case of DCCF and given by

$$\mathbf{h} = \text{Dominant eigenvector}\{\mathbf{S}^{-1}\mathbf{W}\} \quad (90)$$

Given a test input \mathbf{z} , the distances d_k between the transformed input and the ideal shape for class k is computed by using MSE-then-total approach [30]. In this approach, n distances are computed for class k . The j -th distance, d_{jk} , is defined as follows

$$d_{jk} = \left| \mathbf{H}_j^* \mathbf{z}^j - \mathbf{H}_j^* \mathbf{m}_k^j \right|^2 \quad 1 \leq j \leq n \quad (91)$$

The distance d_{jk} can be rewritten as

$$d_{jk} = p_j + b_{jk} - 2\Re\{(\mathbf{z}^j)^+ \mathbf{h}_{jk}\} \quad (92)$$

where

$$p_j = \left| \mathbf{H}_j^* \mathbf{z}^j \right|^2 \quad (93)$$

$$b_{jk} = \left| \mathbf{H}_j^* \mathbf{m}_k^j \right|^2 \quad (94)$$

$$\mathbf{h}_{jk} = \mathbf{H}_j \mathbf{H}_j^* \mathbf{m}_k^j \quad (95)$$

The inner products, shown as the third terms in Equation (92), are between the j -th variation of the input, \mathbf{z}^j and the corresponding j -th filter, \mathbf{h}_{kj} . The total distance d_k to a class is then found by

$$d_k = \sum_{j=1}^n d_{jk} \quad (96)$$

The input image is assigned to the class with the least total distance.

11. Target Tracking in FLIR Imagery

In general, tracking of a moving pattern/target requires recognizing and then locating the target in a scene, finding the target motion, understanding the direction of motion of the target, and then following that target as it moves through the sequence of image frames. The detection and tracking of desired targets in a real life image corrupted by noise, clutter, illumination and other three-dimensional (3D) artifacts, poses a very complex problem and demands sophisticated solutions using pattern recognition and motion estimation methods [31,32]. Things become more complicated if there are more than one target in the scene and simultaneous multiple targets tracking is required.

Forward-looking infrared (FLIR) images are frequently used in automatic target recognition (ATR) applications. It is challenging to detect and track targets in FLIR imagery. To detect independent moving objects in FLIR image sequences, the sensor properties have to be taken into account. Additional challenges are caused due to many important differences of FLIR images with visual sequences [33,34]. Many researchers have investigated various approaches for detection, recognition, classification and pose estimation of targets from FLIR images including both matched spatial filter (MSF) based correlators and joint transform correlators [33–40]. However, the application of MSFs or their variants (e.g., MACH, DCCF) for the FLIR imagery is very limited; although those have been used for the simulated and real synthetic aperture radar (SAR) and laser radar (LADAR) imagery [24,41–47].

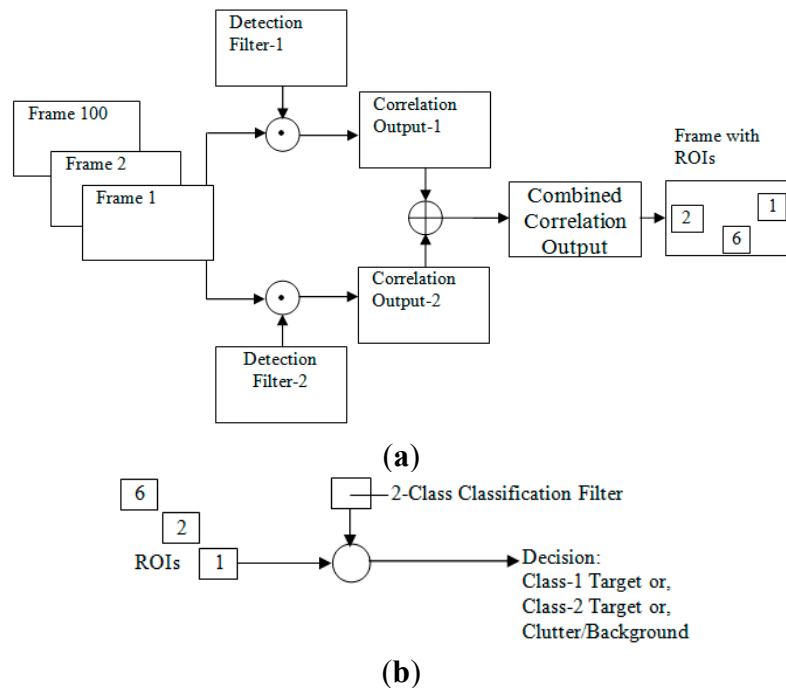
In this section, three different algorithms are demonstrated for pattern recognition and tracking based on the combination of the detection and classification filters [48–51]:

- MACH filter-based detection and DCCF-based classification (MACH-DCCF)
- MACH filter-based detection and PDCCF-based classification (MACH-PDCCF)
- EMACH filter-based detection and PDCCF-based classification (EMACH-PDCCF)

The detection filters are trained by the target images of expected size and orientation variations with expected size of input scene. The classification filters are formulated with the expected size of target images and trained by the target images of expected size and orientation variations. The first step of the real time system is detection, which involves correlating the input scene with all detection filters (one for each desired or expected target class) and combining the correlation outputs. In the second step, a predefined number of ROIs having the expected size of target images are selected based on the regions having higher correlation peak values in the combined correlation output. To ensure that all desired or expected targets are included in the ROIs, the number of ROIs should be at least three times higher than the number of expected targets. Classification filters are then applied to these ROIs and target types along with clutters are identified based on a distance measure and a threshold. Moving target detection and tracking are accomplished by following this technique for all incoming image frames by applying the same filters.

Multiple detection filters and classification filters are formulated for each target based on different size ranges or aspect angles. All of the filters for different ranges can be applied simultaneously through the whole range of the image sequence and decision can be made based on the output of the filter corresponding to the highest correlation peak or minimum distance. However, in this illustration, one detection filter and one classification filter is used for each class of targets for a particular range. A block diagram of the method for real time pattern recognition and tracking is shown in Figure 1 for a two-class detection system.

Figure 1. Schematic diagram of the proposed technique for (a) detection stage; and (b) classification stage.



11.1. Image Dataset

The FLIR image database used in this research is supplied by Army Missile Command (AMCOM). This image database has a total of 50 real life infrared video sequences, some of which contain a single target in the scene and some contain multiple targets. In general, the image sequences are closing sequences, *i.e.*, the targets become closer to the observer as the later frames appear in the scene. Thus, the size and signature of the targets change from the first frame to the last frame. Moreover, as the targets move, there are changes in the targets' orientations from one frame to the next. The database is also associated with ground truth data files containing the list of targets at each frame of each image sequence and their size and location in the frame. Among the 50 sequences, the proposed techniques have been applied for several single and multiple targets image sequences and the techniques are still being tested on other remaining sequences. For this paper, the analysis on 4 single target sequences (L1415, L2018, L2312, M1406), 2 two-target sequences (L1701, L1911) and 1 three-target sequence (L1618) have been reported.

11.2. Single-Target Image Sequences

At first, consider Sequence L2018, which has the highest difficulty level among the selected single target image sequences. In this sequence, there are a total of 448 frames, each of which contains the same target (tank1). The size of this target in the first frame is 3×4 pixels whereas that in the final frame is 12×23 pixels. For effective detection and classification, proper selection of the training images is important. It is obvious that a single detection or classification filter obtained by exploiting the training images from this large variation of sizes may lose its selectivity. So, for this particular sequence, two different detection filters (MACH or EMACH) are formed for the target to use in two ranges of the image frames depending on the size of the target.

The first range detection filters are trained using the target patches taken from the first 200 frames at an interval of 5 frames (the 1st, 5th, 10th, 15th, ..., 200th frames). The ground truth data files are used to read the coordinates and sizes of the targets and then to segment out the target patches from the original frames. Each of these patches is then mean-subtracted and normalized dividing by the RMS value of the mean-subtracted patch. Thereafter, each patch is placed at the center of a 118×118 -pixel zero padded matrix to form a full size training image for the detection filters. A sample training image of this type for Sequence L2018 is shown in Figure 2. The spatial domain representations of the MACH filter and EMACH filter for Range-1, trained by the above mentioned training images, are shown in Figure 3a,b, respectively.

Figure 2. (a) A full size (118×118 -pixel) training image for detection filters created from the target patch of Frame 100 of L2018; and (b) 3D mesh plot of (a).

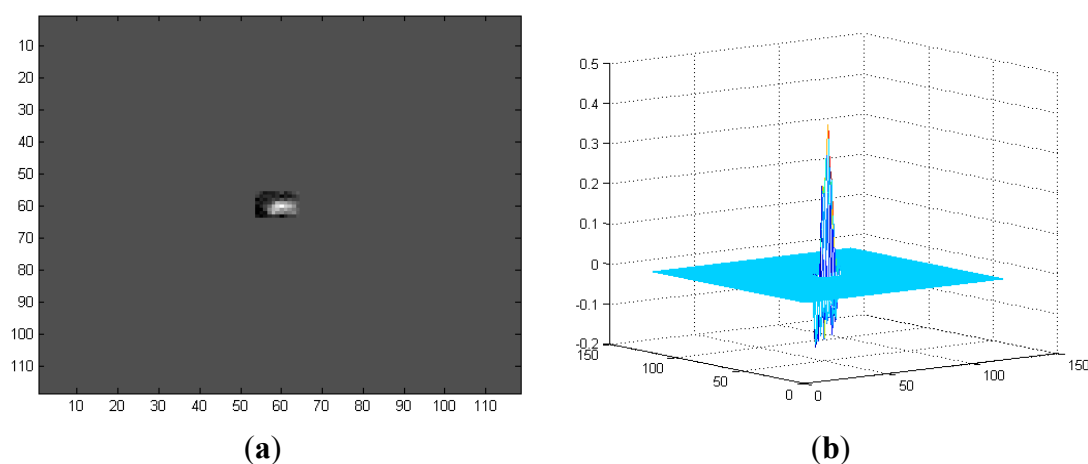
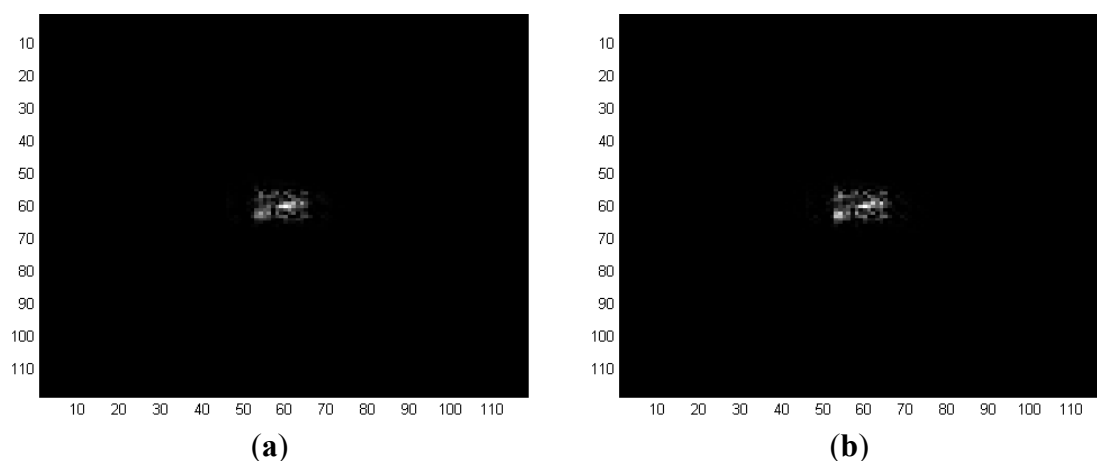
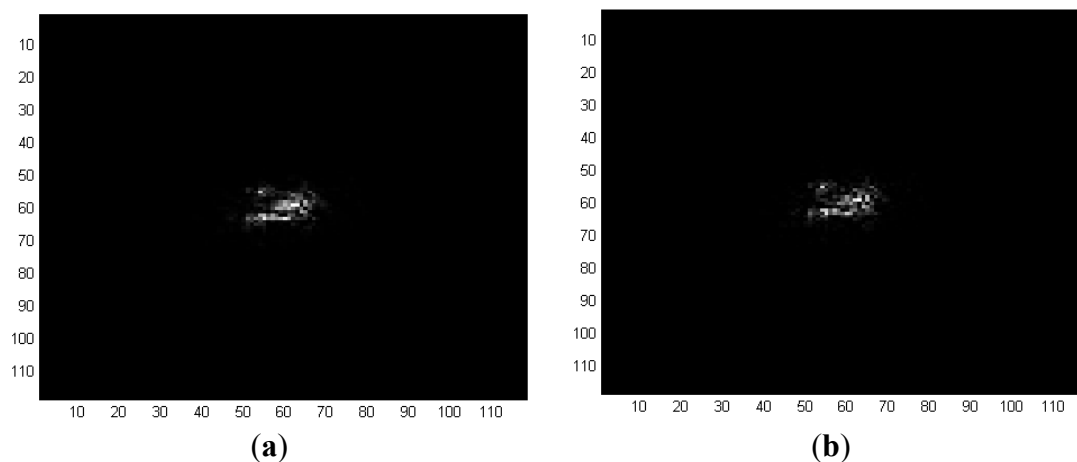


Figure 3. Spatial domain representation of Range-1 detection filters for the target (tank1) of Sequence L2018 (a) MACH filter; and (b) EMACH filter.



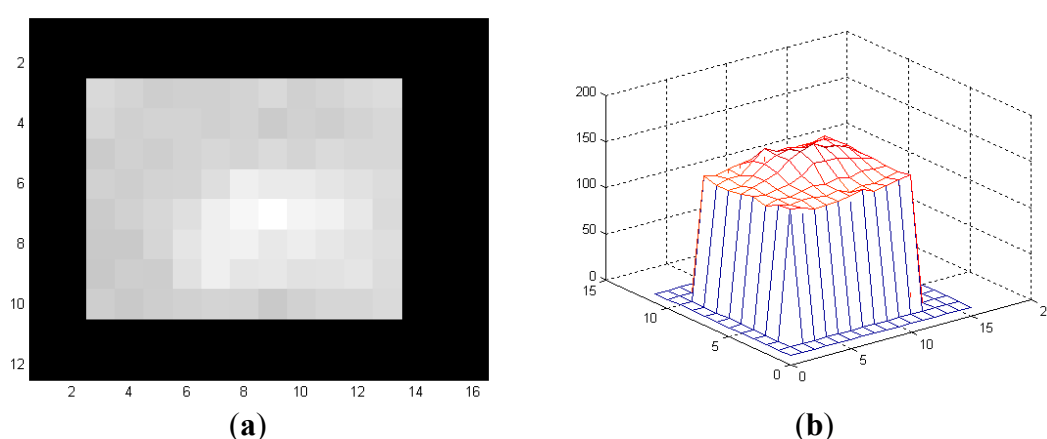
The detection filters for Range-2, to use from Frame 201 to Frame 448, are trained by the target patches taken from the Frames 201 to 300 at an interval of 5 frames. The corresponding Range-2 MACH and EMACH filters are shown in Figure 4a,b, respectively.

Figure 4. Spatial domain representation of Range-2 detection filters for the target (tank1) of Sequence L2018 (a) MACH filter; and (b) EMACH filter.



In general, the classification filters (DCCF or PDCCF) are also required to be formulated for different ranges of the targets for better selectivity. The sizes of these filters are usually chosen as the expected size of the targets in the corresponding range of image frames. Target patches are taken as before but they are not normalized for classification filters. Each patch is placed at the center of a zero padded background having the size of the classification filter. If the patch is larger than the filter size, it is truncated at the sides. For Sequence L2018, it is found that the classification filter (either DCCF or PDCCF) of 12×16 pixels trained by the target patches of Frame 1 to Frame 200 at an interval of 4 frames works well for almost all frames. A 12×16 -pixel sample training image of tank1 of Sequence L2018 for classification filter is shown in Figure 5.

Figure 5. (a) A 12×16 -pixel training image for classification filters created from the target patch of Frame 100 of L2018; and (b) 3D mesh plot of (a).



It is assumed that the single target sequences are known to have only one target of interest. Hence, in this work, single class (1-class) classification filters (DCCF, PDCCF) are formulated for each particular sequence using the corresponding target patches. Although the classification filters actually need to be formulated using at least two classes of training images, in this work, 1-class filters are formulated by using slight modification in the basic formula. However, these 1-class approximated filters have been found to work well in most cases.

The training data or parameters for all the four filters (MACH, EMACH, DCCF, PDCCF) used for the three algorithms for detecting and tracking the target (tank1) in Sequence L2018 are summarized in Table 1. The first column of the table represents the name of the target along with the number of frames for which the ground truth data is available for it. The second column shows the filter names with their range indices; and the third column shows the range of the actual image frames where a particular filter is applied. The fourth and fifth columns provide the considered range of the frames and the order or interval of the frames used to take the target patch for training. In Table 1, γ is the filter parameter for MACH; β and γ are the filter parameters for EMACH and Ψ is the set of nonlinear powers used for PDCCF formulation.

Table 1. Training data for Sequence L2018.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	β	γ	Ψ
			Range Taken	Interval Taken				
tank1 (448)	MACH Range-1	1–200	1–200	5	118 × 118	-	0.1	-
	MACH Range-2	201–448	201–300	5	118 × 118	-	1.0	-
	EMACH Range-1	1–200	1–200	5	118 × 118	0.2	0.1	-
	EMACH Range-2	201–448	201–300	5	118 × 118	0.1	1.0	-
-	DCCF	1–448	1–200	4	12 × 16	-	-	-
-	PDCCF	1–448	1–200	4	12 × 16	-	-	1.0, 1.5, 2.0

Filter formulations of the three other single target sequences (L1415, L2312 and M1406) are also similar to Sequence L2018. To overcome the size variation from the initial frames to the final frames, multiple filters of multiple range bins are required in some cases to apply at different ranges. The training data for these three single target sequences are depicted in Tables 2–4, respectively.

Table 2. Training data for Sequence L1415.

Target (No. of Frames)	Filter	Working Frame Range	Training frames		Filter Size	β	γ	Ψ
			Range Taken	Interval Taken				
Mantruck (281)	MACH	1–281	1–100	5	118 × 118	-	1	-
	EMACH	1–281	1–100	5	118 × 118	0.1	1	-
-	DCCF	1–281	1–100	5	12 × 16	-	-	-
-	PDCCF	1–281	1–100	5	6 × 8	-	-	1.0, 1.5, 2.0

Table 3. Training data for Sequence L2312.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	β	γ	Ψ
			Range Taken	Interval Taken				
APC1 (368)	MACH	1–368	1–100	5	118 × 118	-	1	-
	EMACH	1–368	1–300	10	118 × 118	0.1	1	-
-	DCCF	1–368	1–300	10	12 × 16	-	-	-
-	PDCCF	1–368	1–300	10	8 × 12	-	-	1.0, 1.5, 2.0

Table 4. Training data for Sequence M1406.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	β	γ	Ψ
			Range Taken	Interval Taken				
Bradley (380)	MACH	1–380	1–100	5	118×118	-	1	-
	EMACH	1–380	1–100	5	118×118	0.1	1	-
-	DCCF	1–380	1–300	10	14×16	-	-	-
-	PDCCF Range-1	1–200	1–100	5	8×10	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	201–380	201–300	5	8×10	-	-	1.0, 1.5, 2.0

For analysis, consider the first frame (Frame 1) of Sequence L2018 shown in Figure 6a, apply Fourier transform to it and then correlate with the Fourier domain filter MACH Range-1 to obtain the correlation output of Figure 6b. It is obvious that the correlation output contains false peaks and high-energy diffractions at low frequency which make the actual correlation signal negligible. For this reason, it requires 33 ROIs to include the target of interest as shown in Figure 7b.

Figure 6. (a) Frame 1 of Sequence L2018; and (b) correlation output with the filter maximum average correlation height (MACH) Range-1.

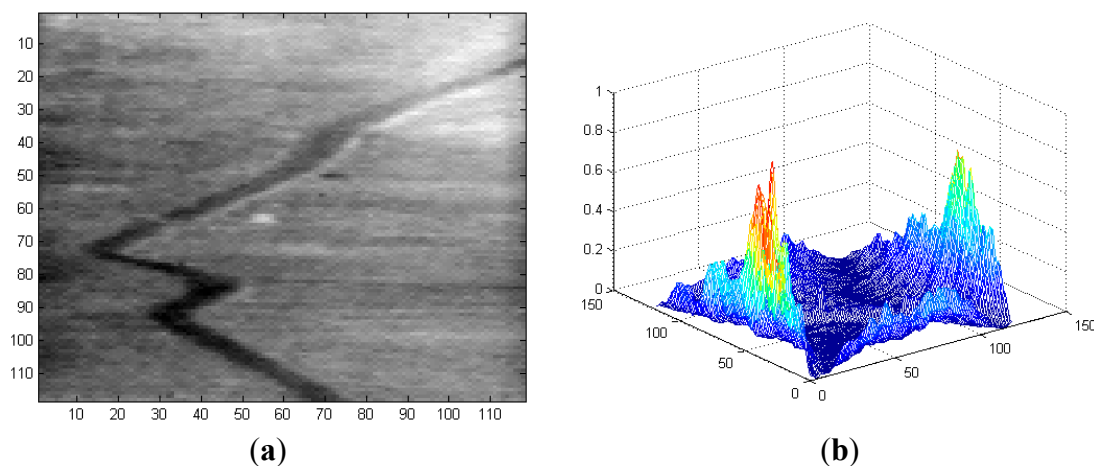
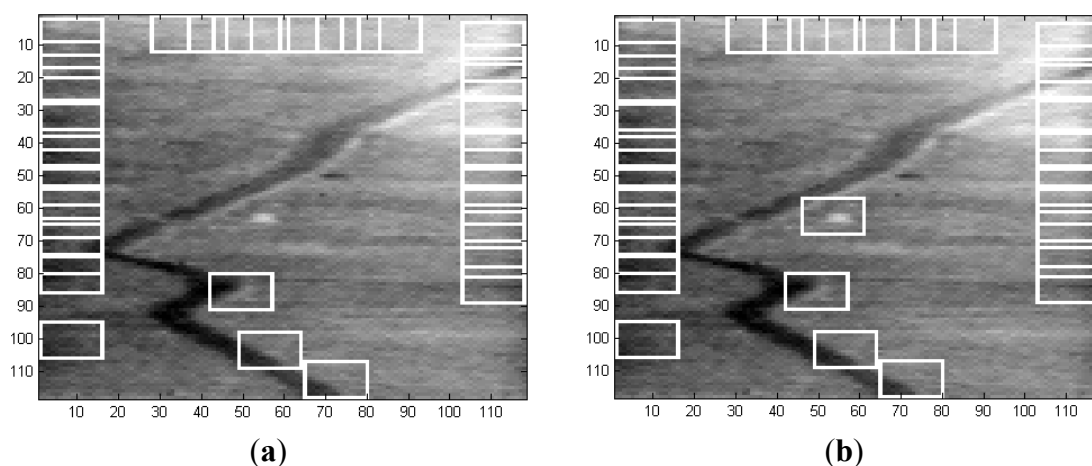
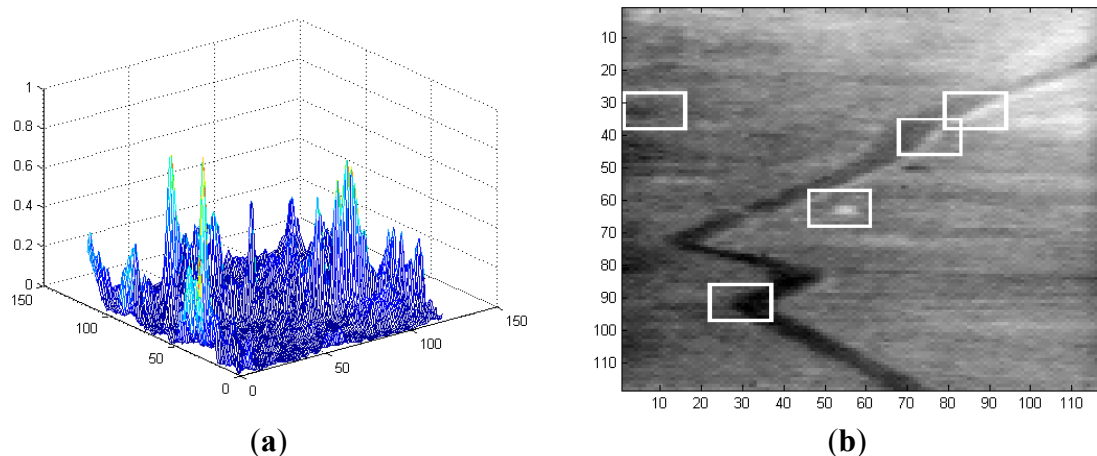


Figure 7. (a) 32 ROIs in Frame 1 of Sequence L2018; and (b) 33 ROIs in Frame 1 of Sequence L2018.



To eliminate the strong low frequency components, a notch filter is used before applying the inverse Fourier transform operation, which actually suppresses the Fourier domain components along the axes to zero. The correlation output after applying this notch filter is shown in Figure 8a. Five ROIs in Frame 1, selected based on this correlation output, are shown in Figure 8b where the target (tank1) is included within these ROIs. This notch filter is used in the detection stage of each frame using either MACH or EMACH filter for all image sequences analyzed in this work. The detection results using EMACH filter are similar to MACH filter with or without the use of notch filter.

Figure 8. (a) Correlation output for Frame 1 of Sequence L2018 using MACH Range-1 filter along with notch filter; and (b) 5 ROIs in Frame 1.



It is obvious that the detection and classification cannot be done correctly with detection filters alone. Because of the presence of multiple identical and different targets and clutters, the highest correlation peak or PSR value is not always guaranteed for the desired target. Therefore, classification filters are used for improved discrimination for the single target image sequences too. It is found that the 1-class classification filters work well in rejecting the clutters and backgrounds. However, to make a decision for identifying the target and rejecting the clutter the ROI having the least distance is considered as the potential target. If the distance to the ROI having the second minimum distance is not higher than the prescribed percentage of the least distance, then the ROI having the least distance is rejected as clutter or background. Since EMACH filter has improved clutter rejection capability, the application of EMACH filter in the detection stage facilitates in lowering the number of ROIs introduced into the classification stage. For this particular sequence (L2018), it is found in the simulation that 8 ROIs are required to include the target in almost all frames in the case of MACH filter, whereas 6 ROIs are sufficient with EMACH filter. However, for generality, 8 ROIs are also considered in case of EMACH filter for this sequence.

All single target sequences are tested by using the three developed algorithms: MACH-DCCF, MACH-PDCCF and EMACH-PDCCF. Some of the frames of Sequence L2018 showing the results after applying the EMACH-PDCCF algorithm are given in Figure 9. The tracking algorithm inserts a “T” mark at the locations of the detected targets (tank1) at each frame as shown in Figure 9. The number at the lower left corner of each frame indicates the frame number. In Figure 9, some sample frames are also included where the classification is incorrect or no decision is inferred.

The detection and tracking results for all single target sequences are summarized for all the three algorithms in Table 5. In the threshold column of the table, a single value indicates that a single threshold is used for all ranges. From Table 5, it is observed that the poorest performance is achieved for Sequence L2018 among the four single target sequences presented. This may be attributed to the scene complexities and the drawback of using same classification filters through all the ranges of the sequence.

Figure 9. Target detection and classification results of Sequence L2018.

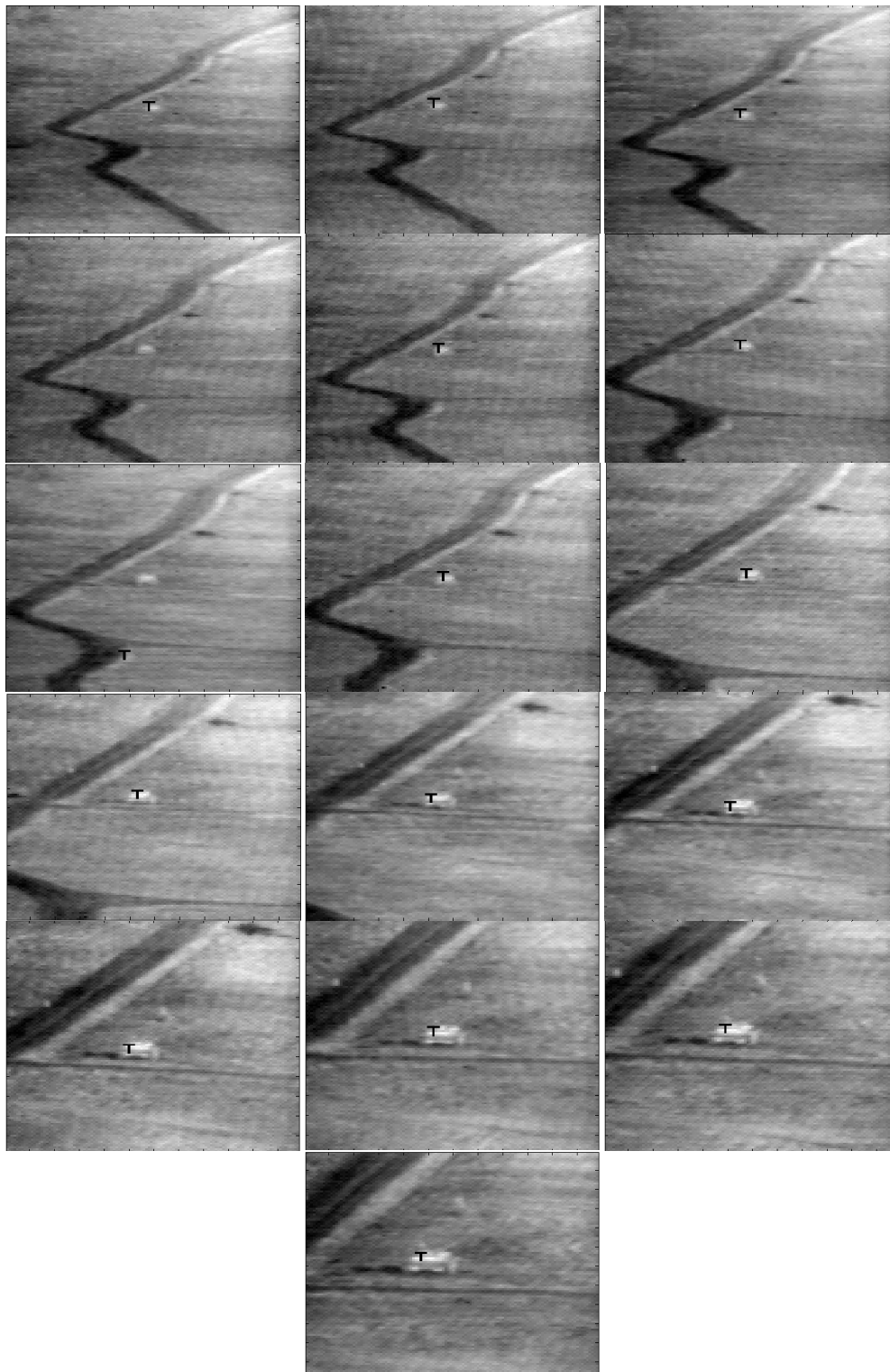


Table 5. Tracking results of single-target sequences.

Seq. Name	Total Frames	No. of ROIs Taken	Threshold	Target Name	No. of Frames Target Present	No. of Frames Detected Correctly	Total No. of False Alarms	Percentage of Successful Detection
MACH-DCCF Algorithm								
L1415	281	4	1.00	mantruck	281	281	0	100
L2018	448	8	0.99	tank1	448	357	36	80
L2312	368	4	1.00	APC1	368	366	2	99
M1406	380	4	1.00	Bradley	380	380	0	100
MACH-PDCCF Algorithm								
L1415	281	4	1.00	mantruck	281	263	9	94
L2018	448	8	0.99	tank1	448	371	37	83
L2312	368	4	1.00	APC1	368	368	0	100
M1406	380	4	1.00	Bradley	380	353	27	93
EMACH-PDCCF Algorithm								
L1415	281	4	1.00	mantruck	281	274	1	98
L2018	448	8	1.00	tank1	448	386	60	86
L2312	368	4	1.00	APC1	368	368	0	100
M1406	380	4	1.00	Bradley	380	354	26	93

11.3. Two-Target Image Sequences

To assess the performance of the algorithms with two targets present in the scene, consider Sequences L1701 and L1911. The two targets in Sequence L1701 are Bradley and pickup. Out of 388 frames of this sequence, ground truth data for Bradley is available for 371 frames and that for pickup is available for 43 frames. The pickup disappears from the scene at Frame 31 and reappears at Frame 81 and again disappears at a later frame. The two targets, APC1 and tank1, in Sequence L1911 are present in all the 165 frames of the sequence. In these image sequences, the target sizes increase significantly from the first frame to the final frame. Therefore, different filters must be used for different ranges. For a particular range, two detection filters (MACH or EMACH) and a 2-class classification filter (DCCF or PDCCF) are required for a two-target sequence. Table 6 shows the training data/parameters selected for different filters for Sequence L1701 while Table 7 displays the same parameters for Sequence L1911. It may be mentioned that even if one target disappears after a few frames, both the detection filters and the 2-class classification filter are continued to apply to the remaining image frames; because in reality, it is not known whether any target is going out or coming back. This ensures the detection of a target that may reappear after a few frames.

Table 6. Training data for Sequence L1701.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	β	γ	Ψ
			Range Taken	Interval Taken				
Bradley (371)	MACH Range-1	1–100	1–100	5	118×118	-	1.0	-
	MACH Range-2	101–200	101–200	5	118×118	-	1.0	-
	MACH Range-3	201–300	201–300	5	118×118	-	0.1	-
	MACH Range-4	301–388	301–370	5	118×118		0.1	-
	EMACH Range-1	1–100	1–100	5	118×118	0.1	1.0	-
	EMACH Range-2	101–200	101–200	5	118×118	0.1	1.0	-
	EMACH Range-3	201–300	201–300	5	118×118	0.1	0.1	-
	EMACH Range-4	301–388	301–370	5	118×118	0.1	0.1	-
Pickup (43)	MACH Range-1	1–388	1–43	3	118×118		1.0	-
	EMACH Range-1	1–388	1–43	3	118×118	0.1	1.0	-
-	DCCF Range-1	1–100	1–100	5	6×8	-	-	-
-	DCCF Range-2	101–200	201–300	5	8×12	-	-	-
-	DCCF Range-3	201–300	201–300	5	10×18	-	-	-
-	DCCF Range-4	301–388	301–370	5	10×18	-	-	-
-	PDCCF Range-1	1–100	1–100	5	8×8	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	101–200	101–200	5	8×12	-	-	1.0, 1.5, 2.0
-	PDCCF Range-3	201–300	201–300	5	10×18	-	-	1.0, 1.5, 2.0
-	PDCCF Range-4	301–388	301–370	5	10×18	-	-	1.0, 1.5, 2.0

Table 7. Training data for Sequence L1911.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	β	γ	Ψ
			Range Taken	Interval Taken				
APC1 (165)	MACH Range-1	1:100	1:100	5	118×118	-	0.1	-
	MACH Range-2	101:165	101:150	5	118×118	-	0.1	-
	EMACH Range-1	1:100	1:100	5	118×118	0.1	0.1	-
	EMACH Range-2	101:165	101:150	5	118×118	0.1	0.1	-
tank1 (165)	MACH Range-1	1:100	1:100	5	118×118	-	0.1	-
	MACH Range-2	1:165	101:150	5	118×118	-	0.1	-
	EMACH Range-1	1:100	1:100	5	118×118	0.1	0.1	-
	EMACH Range-2	1:165	101:150	5	118×118	0.1	0.1	-
-	DCCF Range-1	1:100	1:100	5	8×16	-	-	-
-	DCCF Range-2	101:130	101:130	5	8×16	-	-	-
-	DCCF Range-3	131:165	131:160	5	16×36	-	-	-
-	PDCCF Range-1	1:130	1:100	5	8×16	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	1:130	1:100	5	8×16	-	-	1.0, 1.5, 2.0
-	PDCCF Range-3	131:165	131:160	5	16×36	-	-	1.0, 1.5, 2.0

Using the detection and classification filters having the design parameters in Tables 6 and 7, all algorithms are tested for detection, classification and tracking of the objects in Sequences L1701 and L1911. The results obtained after applying the EMACH-PDCCF algorithm on Sequence L1701 are shown for some sample frames in Figure 10. The tracking algorithm inserts “T1” for the detected Class-1 type targets (Bradley) and “TII” for the detected Class-2 type targets (pickup) at their

corresponding location in the image frame. The frame number is shown at the lower left corner of each frame displayed. From Figure 10, it is obvious that the tracking algorithm can successfully detect and classify the targets when they are present in the input scene. The complete tracking results for both the sequences are summarized in Table 8 for all algorithms. In the threshold column of the table, a single value indicates that a single threshold is used for all ranges. Otherwise, different threshold values in that column displays the thresholds used for various ranges of classification filters employed. It is observed that MACH-DCCF algorithm fails for Sequence L1911 in detecting and classifying the targets, and rejecting the clutters. On the other hand, EMACH-PDCCF algorithm provides the best results considering all the factors simultaneously, such as, required number of ROIs, percentage of successful detection and the total number of false alarms.

Figure 10. Target detection and classification results of Sequence L1701.

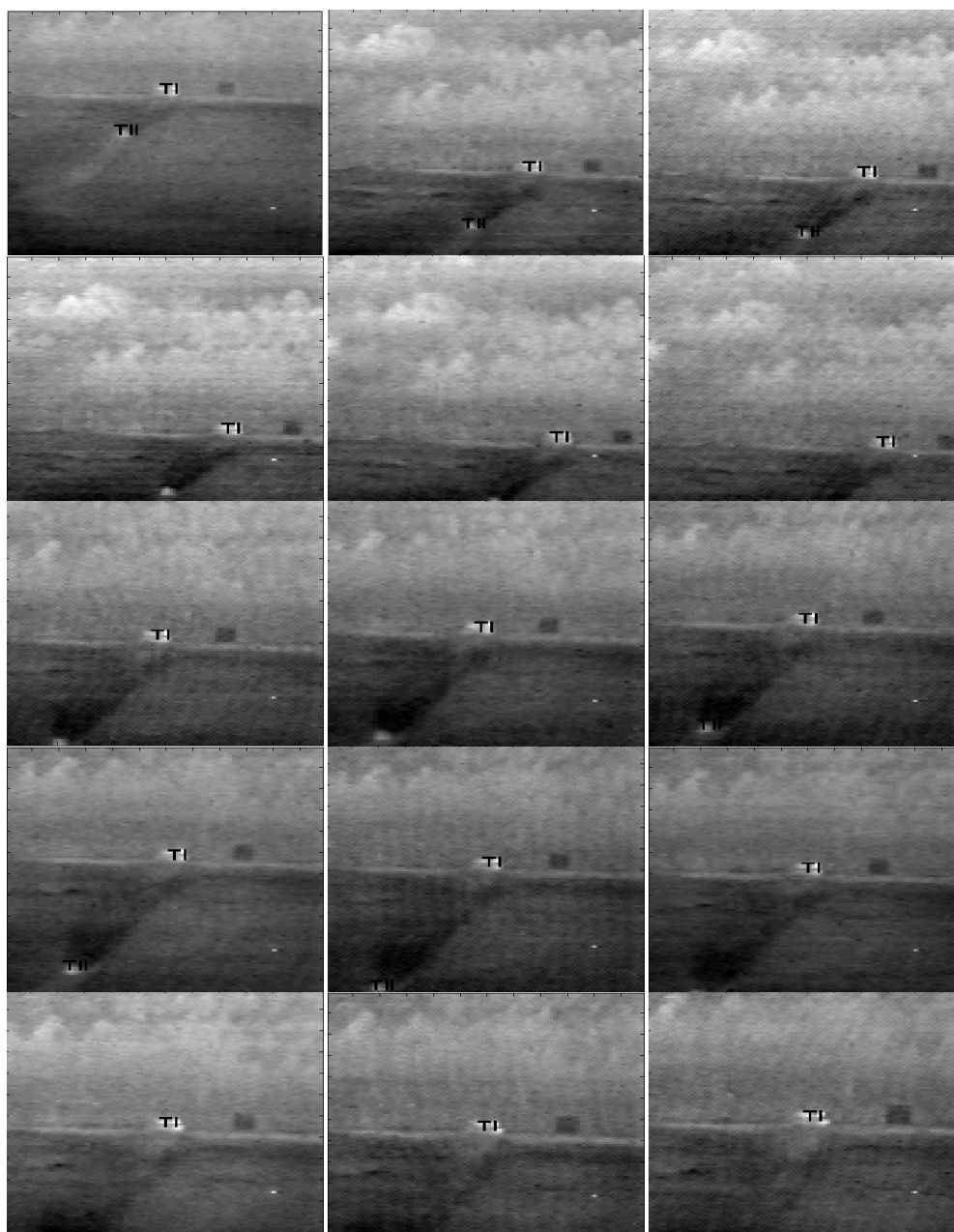


Figure 10. Cont.

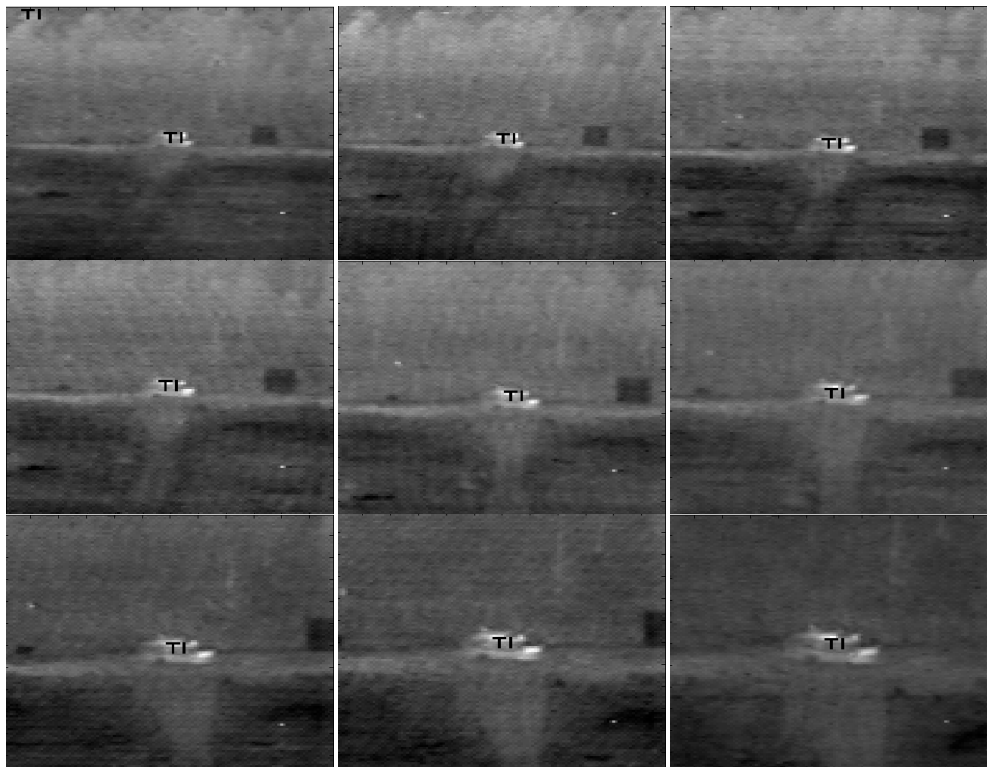


Table 8. Tracking results of two-target sequences.

Seq. Name	Total Frames	No. of ROIs Taken	Threshold	Target Name	No. of Frames Target Present	No. of Frames Detected Correctly	Total No. of False Alarms	Percentage of Successful Detection
MACH-DCCF Algorithm								
L1701	388	6	0.70, 0.30, 0.05, 0.04	Bradley	388	230	248	59
				pickup	45	26	2	58
L1911	165	6	0.40	APC1	-	-	-	Fails
				Tank1	-	-	-	Fails
MACH-PDCCF Algorithm								
L1701	388	6	0.40, 0.05, 0.06, 0.06	Bradley	388	369	12	95
				pickup	45	28	1	62
L1911	165	6	0.40	APC1	165	152	0	92
				tank1	165	165	3	100
EMACH-PDCCF Algorithm								
L1701	388	4	0.40, 0.05, 0.06, 0.06	Bradley	388	369	9	95
				pickup	45	31	8	69
L1911	165	4	0.40	APC1	165	160	0	97
				tank1	165	165	5	100

11.4. Three-Target Image Sequences

The three target images in Sequence L1618 are APC1, M60 and truck. Out of 300 frames in this sequence, ground truth data for APC1 is available for 291 frames, ground truth data for M60 is found

for 101 frames and that for truck is found for 6 frames. The truck disappears from the sequence at Frame 18 and M60 disappears from the sequence at Frame 104. Like the other sequences, the size of the targets increases significantly from the first frame to the final frame. Thus, different filters are required for different ranges of this sequence. Assuming three expected targets in the scene for any particular range, three detection filters (MACH or EMACH) and a three-class classification filter (DCCF or PDCCF) are required for every frame in each range of this sequence. The training data/parameters for different filters, chosen for Sequence L1618 are displayed in Table 9.

Table 9. Training data for Sequence L1618.

Target (No. of Frames)	Filter	Working Frame Range	Training Frames		Filter Size	β	γ	Ψ
			Range Taken	Interval Taken				
APC1 (291)	MACH Range-1	1–100	1–100	5	118×118	-	1.0	-
	MACH Range-2	101–200	101–200	5	118×118	-	1.0	-
	MACH Range-3	201–300	201–290	5	118×118	-	1.0	-
	EMACH Range-1	1–100	1–100	5	118×118	0.1	1.0	-
	EMACH Range-2	101–200	101–200	5	118×118	0.1	1.0	-
	EMACH Range-3	201–300	201–290	5	118×118	0.1	1.0	-
M60 (101)	MACH Range-1	1–300	1–100	5	118×118	-	1.0	-
	EMACH Range-1	1–300	1–100	5	118×118	0.1	1.0	-
Truck (6)	MACH Range-1	1–300	1–6	1	118×118	-	1.0	-
	EMACH Range-1	1–388	1–6	1	118×118	0.1	1.0	-
-	DCCF Range-1	1–100	1–100	5	6×12	-	-	-
-	DCCF Range-2	101–200	201–300	5	8×16	-	-	-
-	DCCF Range-3	201–300	201–290	5	8×16	-	-	-
-	PDCCF Range-1	1–100	1–100	5	6×12	-	-	1.0, 1.5, 2.0
-	PDCCF Range-2	101–200	101–200	5	8×16	-	-	1.0, 1.5, 2.0
-	PDCCF Range-3	201–300	201–290	5	8×16	-	-	1.0, 1.5, 2.0

To evaluate the performance of all the algorithms for detection, clutter rejection, and classification as well as tracking of the three objects in three-target sequence, the designed detection and classification filters of Table 9 for different ranges are applied to Sequence L1618. The results obtained for the EMACH-PDCCF algorithm with Sequence L1618 are shown for some sample frames in Figure 11. The tracking algorithm places “T1” for the detected Class-1 type targets (APC1), “TII” for the detected Class-2 type targets (M60) and “TIII” for the detected Class-3 type targets (truck) at their corresponding locations in the frames. In Figure 11, the frame numbers are shown at the lower left corner of each frame. The complete tracking results for the sequence are summarized in Table 10 for all algorithms. In the threshold column of the table, a single value indicates that a single threshold is used for all ranges. Otherwise, different threshold values in that column displays the thresholds used for various ranges of classification filters employed.

Figure 11. Target detection and classification results of Sequence L1618.

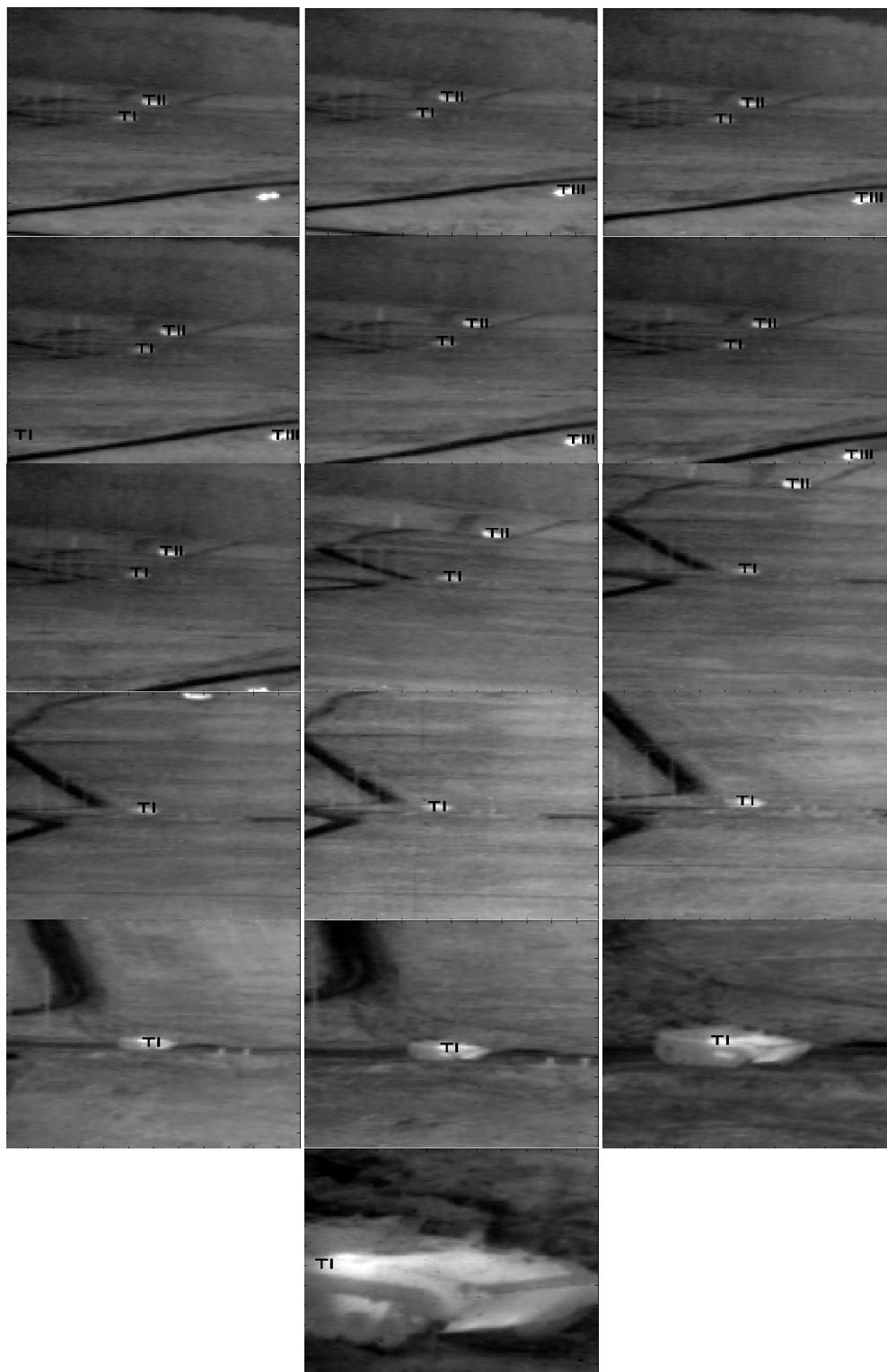


Table 10. Tracking results of three-target sequences.

Seq. Name	Total Frames	No. of ROIs Taken	Threshold	Target Name	No. of Frames Target Present	No. of Frames Detected Correctly	Total No. of False Alarms	Percentage of Successful Detection
MACH-DCCF Algorithm								
L1618	300	6	0.30, 0.52, 0.52	APC1	300	273	54	91
				M60	103	71	0	69
				truck	17	0	0	0
MACH-PDCCF Algorithm								
L1618	300	6	0.52	APC1	300	297	20	99
				M60	103	99	0	96
				truck	17	12	0	71
EMACH-PDCCF Algorithm								
L1618	300	4	0.52	APC1	300	297	23	99
				M60	103	99	0	96
				truck	17	12	0	71

12. Conclusions

Pattern recognition and tracking in FLIR imagery is a challenging problem due to various factors such as low resolution, low signal-to-noise ratio, different 3D orientations of the targets, effects of global motion, and close proximity with similar objects. In this paper, we reviewed the recent trends and advancements in distortion-invariant pattern recognition algorithms for single/multiple target detection and tracking in FLIR imagery using correlation filters. Each detection/tracking algorithm utilizes various properties of targets and image frames of a given FLIR sequence. Test results using real life FLIR image sequences are presented to verify the effectiveness of the filter-based pattern recognition and tracking techniques. Future work in this area would include a review of techniques beyond correlation that are particularly useful for high resolution targets. Also, development and inclusion of techniques with a dynamic update of the target model may be considered.

Conflicts of Interest

The authors declare no conflict of interest.

References

- Schalkoff, R. *Pattern Recognition, Statistical, Structural and Neural Approaches*; John Wiley & Sons: New York, NY, USA, 1992.
- Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*, 2nd ed.; Wiley-Interscience: New York, NY, USA, 2000.
- Gonzalez, R.C.; Woods, R.E.; *Digital Image Processing*, 2nd ed.; Prentice-Hall, Inc.: Taiwan, 2002.
- Brunndstrom, K.; Schenkman, B.N.; Jacobson, B. Object detection in cluttered infrared images. *Opt. Eng.* **2003**, *42*, 388–399.

5. Strehl, A.; Aggarwal, J.K. Detecting moving objects in airborne forward looking infra-red sequences. *J. Mach. Vis. Appl.* **2000**, *11*, 267–276.
6. Yilmaz, A.; Shafique, K.; Shah, M. Target tracking in airborne forward looking infrared imagery. *Image Vis. Comput. J.* **2003**, *21*, 623–635.
7. Alam, M.S.; Haque, M.; Khan, J.F.; Kettani, H. Fringe-adjusted JTC based target detection and tracking in FLIR image sequence. *Opt. Eng.* **2004**, *43*, 1407–1413.
8. VanderLugt, A. Signal detection by complex spatial filtering. *IEEE Trans. Inf. Theory* **1964**, *10*, 139–146.
9. Casasent, D.; Furman, A. Sources of correlation degradation. *Appl. Opt.* **1977**, *16*, 1652–1661.
10. Gianino, P.D.; Horner, J.L. Phase-only matched filtering. *Appl. Opt.* **1984**, *23*, 812–816.
11. Mu, G.G.; Wang, X.M.; Wang, Z.Q. Amplitude-compensated matched filtering. *Appl. Opt.* **1988**, *27*, 3461–3463.
12. Awwal, A.A.S.; Karim, M.A.; Jahan, S.R. Improved correlation discrimination using an amplitude modulated phase-only filter. *Appl. Opt.* **1990**, *29*, 233–236.
13. Hester, C.F.; Casasent, D. Multivariant technique for multiclass pattern recognition. *Appl. Opt.* **1980**, *19*, 1758–1761.
14. Alam, M.S.; Awwal, A.A.S. Scale invariant amplitude modulated phase-only filtering. *J. Opt. Laser Technol.* **2000**, *32*, 231–234.
15. Alam, M.S.; Chen, X.; Karim, M.A. Distortion invariant fringe-adjusted joint transform correlator. *J. Appl. Opt.* **1997**, *36*, 7422–7427.
16. Casasent, D. Unified synthetic discriminant function computational formulation. *Appl. Opt.* **1984**, *23*, 1620–1627.
17. Bahri, Z.; Kumar, B.V.K.V. Generalized synthetic discriminant functions. *J. Opt. Soc. Am. A* **1988**, *5*, 562–571.
18. Kumar, B.V.K.V. Minimum variance synthetic discriminant function. *J. Opt. Soc. Am. A* **1986**, *3*, 1579–1584.
19. Mahalanobis, A.; Kumar, B.V.K.V.; Cassasent, D. Minimum average correlation energy filters. *Appl. Opt.* **1987**, *26*, 3633–3640.
20. Casasent, D.; Ravichandran, G.; Bollapraggada, S. Gaussian MACE correlation filters. *Appl. Opt.* **1991**, *30*, 5176–5181.
21. Kumar, B.V.K.V.; Mahalanobis, A.; Song, S.; Sims, S.R.F.; Epperson, J.F. Minimum squared error synthetic discriminant functions. *Opt. Eng.* **1992**, *31*, 915–922.
22. Mahalanobis, A.; Kumar, B.V.K.V.; Sims, S.R.F.; Epperson, J. Unconstrained correlation filters. *Appl. Opt.* **1994**, *33*, 3751–3759.
23. Mahalanobis, A.; Kumar, B.V.K.V. Optimality of the maximum average correlation height filter for detection of targets in noise. *Opt. Eng.* **1997**, *36*, 26423–2648.
24. Alkanhal, M.; Kumar, B.V.K.V.; Mahalanobis, A. Improved clutter rejection in automatic target recognition (ATR) synthetic aperture radar (SAR) imagery using the extended maximum average correlation height (EMACH) filter. *Proc. SPIE.* **2000**, *4053*, 332–339.
25. Alkanhal, M.; Kumar, B.V.K.V.; Mahalanobis, A. Improving the false alarm capabilities of the maximum average correlation height correlation filter. *Opt. Eng.* **2000**, *39*, 1133–1141.

26. Mahalanobis, A.; Carlson, D.W.; Kumar, B.V.K.V.; Sims, S.R.F. Distance classifier correlation filters. *Proc. SPIE* **1994**, *2238*, 2–13.
27. Mahalanobis, A.; Kumar, B.V.K.V.; Sims, S.R.F. Distance classifier correlation filters for distortion tolerance, discrimination and clutter rejection. *Proc. SPIE* **1993**, *2026*, 325–335.
28. Mahalanobis, A.; Kumar, B.V.K.V.; Sims, S.R.F. Distance classifier correlation filters for multiclass target recognition. *Appl. Opt.* **1996**, *35*, 3127–3133.
29. Alkanhal, M.; Kumar, B.V.K.V. Polynomial distance classifier correlation filter for pattern recognition. *Appl. Opt.* **2003**, *42*, 4688–4708.
30. Muise, R.; Mahalanobis, A.; Mohapatra, R.; Li, X.; Han, D.; Mikhael, W. Constrained quadratic correlation filters for target detection. *Appl. Opt.* **2004**, *43*, 304–314.
31. Hwang, J.; Ooi, Y.; Ozawa, S. Visual feedback control system for tracking and zooming a target. *Proc. Int. Conf. Power Electron. Motion Control IEEE* **1992**, *2*, 740–745.
32. Lipton, A.J.; Fujiyoshi, H.; Patil, R.S. Moving target classification and tracking from real-time video. In Proceedings of the Workshop of the Application of Computer Vision, Princeton, NJ, USA, 19–21 October 1998; pp. 8–14.
33. Bal, A.; Alam, M.S. Automatic Target tracking in FLIR image sequences using intensity variation function and template modeling. *IEEE Trans. Instrum. Meas.* **2005**, *54*, 1846–1852.
34. Dawoud, A.; Alam, M.S.; Bal, A.; Loo, C. Target tracking in infrared imagery using weighted composite reference function based decision fusion. *IEEE Trans. Image Process.* **2006**, *15*, 404–410.
35. Alam, M.S.; Bal, A.; Horache, E.; Goh, S.F.; Loo, C.; Regula, S.; Sharma, A. Metrics for evaluating the performance of joint transform correlation based target recognition and tracking algorithms. *Opt. Eng.* **2005**, *44*, 067005.
36. Bal, A.; Alam, M. Dynamic target tracking using fringe-adjusted joint transform correlation and template matching. *Appl. Opt.* **2004**, *43*, 4874–4881.
37. Dawoud, A.; Alam, M.S.; Bal, A.; Loo, C. Decision fusion algorithm for target tracking in infrared imagery. *Opt. Eng.* **2005**, *44*, doi:10.1117/1.1844534.
38. Alam, M.S.; Khan, J.; Bal, A. Heteroassociative multiple-target tracking by fringe-adjusted joint transform correlation. *Appl. Opt.* **2004**, *43*, 358–365.
39. Mahalanobis, A. Correlation filters for object tracking target re-acquisition and smart aimpoint selection. *Proc. SPIE* **1997**, *3073*, 25–32.
40. Mahalanobis, A.; Muise, R. Advanced detection and correlation based automatic target detection. *Proc. SPIE* **2001**, *4379*, 466–471.
41. Mahalanobis, A.; Carlson, D.W.; Kumar, B.V.K.V. Evaluation of MACH and DCCF correlation filters for SAR ATR using MSTAR public data base. *Proc. SPIE* **1998**, *3370*, 460–468.
42. Mahalanobis, A.; Ortiz, L.A.; Kumar, B.V.K.V. Performance of the MACH filter and DCCF algorithms on the 10-class public release MSTAR data set. *Proc. SPIE* **1999**, *3721*, 285–289.
43. Carlson, D.W.; Riddle, J.G. Clutter background spectral density estimation for SAR target recognition with composite correlation filters. *Proc. SPIE* **2003**, *5106*, 64–71.
44. Perona, M.T.; Mahalanobis, A.; Zachery, K.N. LADAR automatic target recognition using correlation filters. *Proc. SPIE* **1999**, *3718*, 388–396.
45. Perona, M.T.; Mahalanobis, A. System-level evaluation of LADAR ATR using correlation filters. *Proc. SPIE* **2000**, *4050*, 69–75.

46. Nevel, A.V.; Mahalanobis, A. Comparative study of maximum average correlation height filter variants using ladar imagery. *Opt. Eng.* **2003**, *42*, 541–550.
47. Sims, S.R.F.; Mahalanobis, A. Performance evaluation of quadratic correlation filters for target detection and discrimination in infrared imagery. *Opt. Eng.* **2004**, *43*, 1705–1711.
48. Bhuiyan, S.; Alam, M.S.; Alkanhal, M. A new two-stage correlation based approach for target detection and tracking in FLIR imagery using EMACH and PDCCF filters. *Opt. Eng.* **2007**, *46*, 086401.
49. Islam, M.F.; Alam, M.S. Improved clutter rejection in automatic target recognition and tracking using eigen-extended maximum average correlation height (EEMACH) filter and polynomial distance classifier correlation filter (PDCCF). *Proc. SPIE* **2006**, *6245*, 62450B.
50. Bhuiyan, S.M.A.; Alam, M.S.; Sims, S.R.F. Target detection, classification and tracking using MACH and PDCCF filter combination. *Opt. Eng.* **2006**, *45*, 116401.
51. Bhuiyan, S.; Khan, J.F.; Alam, M.S. Power enhanced extended maximum average correlation height filter for target detection, to appear. In Proceedings of the IEEE SoutheastCon, Lexington, Kentucky, KY, USA, 13–16 March 2014.

© 2014 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).