Evaluation of Infrared Missile Warning Algorithm Suites

Steve Hary, John McKeeman, and Dale Van Cleave

Wright Laboratories, Avionics Directorate, Information Processing Branch, Wright-Patterson AFB, OH 45433-7301

Abstract

Operation Desert Storm highlighted the effectiveness of infrared (IR) missile warning systems to detect and track hostile missile threats. This, in turn, has sparked renewed interest in the development of IR missile warning algorithms and architectures. The Avionics Directorate has funded several IR missile warning algorithm studies and the performance of six algorithm suites is presented in this paper. All suites were evaluated against registered data from two bands of mid-wave IR (MWIR) imagery and are evaluated for their target detection performance without the aid of tracking algorithms.

The effectiveness of each algorithm suite to detect point targets was measured in terms of missed detections and false alarms. The preliminary results show that for targets with low signal to noise ratios (SNR), missile warning algorithm suites which use simple spatial and spectral filters achieve the same performance as more computationally complex algorithm suites.

1. Algorithm Suites

Traditional IR missile warning algorithm suites use three preprocessing stages to extract exceedances from IR image data. Figure 1 shows the data flow between the preprocessing stages. First, the data are spatially filtered to increase the signal to clutter ratio of potential targets in the data. After spatial filtering, the data are deinterleaved and used as input to a spectral filter. The spectral filter uses the original data and the spatially filtered data from two or more color bands to produce a high signal to clutter ratio image, S(x,y).

The spectrally filtered data is input to a background normalizer and thresholder (BNT) where the data is detected and exceedances are passed to a target tracker. The list of exceedances contains the x,y location of each exceedance, and any statistics calculated for the exceedance pixel and its background. A tracker would use the exceedance information to remove false alarms and pass potential targets to the operator.

Figure 1. Data Flow for Infrared Missile Warning Suites.

Table 1. Algorithms used within each algorithm suite.

<table>
<thead>
<tr>
<th>Suite</th>
<th>Spatial Filter</th>
<th>Spectral Filter</th>
<th>BNT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Laplacian</td>
<td>Ratio</td>
<td>Additive</td>
</tr>
<tr>
<td>2</td>
<td>Laplacian</td>
<td>Correlation</td>
<td>Additive</td>
</tr>
<tr>
<td>3</td>
<td>Median</td>
<td>Ratio</td>
<td>Additive</td>
</tr>
<tr>
<td>4</td>
<td>Median</td>
<td>Correlation</td>
<td>Additive</td>
</tr>
<tr>
<td>5</td>
<td>Morphological</td>
<td>Ratio</td>
<td>Additive</td>
</tr>
<tr>
<td>6</td>
<td>Morphological</td>
<td>Correlation</td>
<td>Additive</td>
</tr>
</tbody>
</table>

The following subsections of Section 1 discuss the algorithm suites outlined above. Within each subsection the algorithms used in each filtering stage are detailed. Section 2 discusses the data used to analyze the algorithm suite performance. Section 3 discusses the data collection method and Section 4 presents the results and conclusions.
1.1 Spatial Filtering

Spatial filters enhance the signal to clutter ratio of potential targets in IR imagery through a process known as clutter removal or clutter rejection. Clutter rejection has been the subject of intense research over the past three decades [1,2,3,4]. In 1959, Robinson was the first to use nonlinear spatial filters, known as Robinson filters, to extract point targets from background clutter in IR imagery [1]. In 1979, Otazo and Parenti showed that targets whose extent is less than one pixel (i.e. point targets) could be effectively detected using linear, matched, spatial filters [3]. In 1980, Otazo, Tung and Parenti explored spatial filtering algorithms for detecting targets whose extent was slightly greater than one pixel [4].

Takken et al. developed matched and least mean square (LMS) spatial filters to increase the signal to noise ratio of targets in a single frame of data [5]. Scribner et al. extended this technique in 1989 to three-dimensions [6]. Their results showed that three-dimensional LMS and matched velocity filters perform better than spatial filters, however, these approaches require many banks of velocity filters. In terms of missed detection and false alarm rates, when the target intensity to background noise ratio is low, (i.e SNR = 1) the three-dimensional LMS filters show a performance improvement by a factor of three over two-dimensional spatial filters. For target intensities to background noise ratios of 3, the performance of their three-dimensional filters is an order of magnitude better than two-dimensional spatial filters. The price of this enhanced performance is increased computational complexity and greater memory requirements.

The optimal size and shape of spatial filters used in IR missile warning algorithms is dependent on noise statistics, the expected intensity distribution, and the SNR of the target and background clutter. Typically, spatial filters are 3 x 3, 5 x 5 or 7 x 7 pixels in extent and the operator is typically square. The size and shape of the spatial filter are primary drivers of the filter’s computational complexity. For this research, all spatial filters were 3 x 3 pixels in extent and the filters were square.

The following subsections discuss the spatial filters used in this research.

1.1.1 Laplacian Filter

The Laplacian filter is a standard convolution filter and the output value, B(x,y), for any input pixel, P(x,y), can be defined as:

\[ B(x,y) = 8 \times P(x,y) - [P(x-1,y-1) + P(x-1,y) + P(x-1,y+1) + P(x,y-1) + P(x,y+1) + P(x+1,y-1) + P(x+1,y) + P(x+1,y+1)] \]  

[1]

1.1.2 Median Subtraction Filter

Another spatial filter used in MWIR imagery is the median subtraction filter (MSF) [7,8]. Here, the median value of the eight nearest neighbors of the center pixel is subtracted from the value of the center pixel. Havlicek first reported on the performance of a two-dimensional (eight neighbor) MSF on MWIR imagery in 1988 [7]. Barnett, in 1989, reported on the statistical properties of one-dimensional (three point) MSF’s and compared their performance to spatial (nine point) LMS filters. Barnett’s MSF provided performance rivaling that of the LMS filters for targets in mild clutter, however, it performed poorly in scenes containing severe clutter [8]. Barnett’s filter required twenty-five percent fewer operations per pixel than the LMS spatial filter. The eight point MSF proposed by Havlicek required twice the computational complexity of linear spatial filters, but provided better performance than Barnett’s MSF. In this research, the Havlicek MSF was used as the baseline median filter algorithm.

The MSF ranks orders the eight nearest neighbors of the input pixel \( P(x,y) \) by sorting them from smallest to largest. The ordered pixels are then labeled \( x_0, x_1, x_2, \ldots, x_7 \). For any band of data, the value of the output pixel from the MSF, \( B(x,y) \), is defined as:

\[ B(x,y) = P(x,y) - M(x,y) \]

where \( M(x,y) \) is defined as:

\[ M(x,y) = \frac{1}{8} \sum_{i=0}^{7} x_i \]

The output pixel value is the median value of the eight nearest neighbors of the center pixel subtracted from the value of the center pixel.

1.1.3 Morphological Filter

In 1988, Ockman et al. used high-pass, morphological spatial filters to enhance the signal to noise ratio of MWIR IR Search and Track (IRST) imagery [9]. These non-linear filters reduce the ringing associated with linear spatial filters, but they require more operations than linear spatial filters.

Morphological filtering uses the structure of the background clutter as a part of the clutter removal process. Specifically, a shape operator which approximates the shape of the clutter is convolved with the image. Pixels within the boundaries of the shape operator are used to determine the output. The most commonly used shape operators are a 3 x 3 pixel cross operator and a 3 x 3 pixel square operator. The cross operator uses the center pixel and its four nearest neighbors to compute the output. The square operator uses the center pixel and its eight nearest neighbors to compute the output.
A morphological filter consists of repeated applications of two basic morphological filtering operations: dilations and erosions. Figure 2 shows that the morphological filter used in Algorithm Suites 5 and 6 consists of four successive filtering passes: dilation, erosion, erosion, and dilation, respectively.

![Diagram of morphological filter](image)

**Figure 2. Flow diagram for morphological filter.**

As an example, consider a dilation using a square shape operator. Let the values of the center pixel \( P(x,y) \) and its eight nearest neighbors be defined as: \( x_0, x_1, x_2, \ldots, x_8 \). The value of the output pixel from the dilation, \( D(x,y) \), is defined as:

\[
D(x,y) = \max\{x_0, x_1, \ldots, x_8\} \tag{3}
\]

An erosion replaces the center pixel with the smallest pixel value from the pixels within the boundaries of the shape operator. Consider an erosion using a cross shape operator. Let the values of the center pixel \( P(x,y) \) and its four nearest neighbors be defined as: \( x_0, x_1, x_2, \ldots, x_4 \). Then, the value of the output pixel from the erosion, \( E(x,y) \), is defined as:

\[
E(x,y) = \min\{x_0, x_1, \ldots, x_4\} \tag{4}
\]

Figure 2 shows that the morphological filter first dilates the pixels in the input image, \( P(x,y) \). The dilated output image is then eroded in two sequential erosion passes. The eroded image is then dilated in the fourth pass of the morphological filter.

Since the morphological filter shown in Figure 2 is a low-pass filter, its final output is subtracted from the input image to provide the high-pass filter used in this research. If any band of data, the value of the output pixel from the morphological filter, \( B(x,y) \), is defined as:

\[
B(x,y) = P(x,y) - E(x,y) \tag{5}
\]

where \( E(x,y) \) is output from the last dilation pass.

### 1.2 Spectral Filtering

Spectral filters further enhance the signal to clutter ratio of potential threats in IR imagery by using target specific emissions in two or more color bands. Missile and aircraft threat emit large amounts of radiation in the 4.2-4.8 \( \mu \)m spectrum. Within this band, however, radiation emitted in the 4.4-4.6 \( \mu \)m (CO\textsubscript{2}) band is completely absorbed by the atmosphere. Therefore, when discriminating hostile missile threats, radiation in the 4.6-4.8 \( \mu \)m band is considered the primary band. A secondary band of interest is the 3.85-4.15 \( \mu \)m region. Other bands of interest within the spectral bandwidth (3.0 - 5.0 \( \mu \)m) of an InSb focal plane array (FPA) include guard bands around the primary and secondary bands. In the following discussion, radiation in the 3.85-4.15 \( \mu \)m region will be called Band 1, and radiation in the 4.6-4.8 \( \mu \)m region will be called Band 2.

Spectral filters assume that energy emitted in Band 2 by missiles is much greater than energy emitted in Band 1. To discriminate between missiles and clutter, spectral filters also assume that the radiation from background clutter varies little between bands. Spectral filters use these characteristics to enhance the signal to clutter ratio of potential missile threats.

Spectral filtering algorithms have been proposed. The simplest algorithms subtract the energy in the secondary and guard bands from the energy in the primary band. An analog implementation for this spectral filter has been proposed for on-sensor plane processing systems and digital implementations (frame subtraction) have been proposed by many sources. Other spectral filter algorithms use a 3 x 3 pixel window around each pixel in multiple bands to generate a factor, \( \gamma \), which is an estimate of the energy ratios or correlation between the backgrounds in the primary and secondary bands. After \( \gamma \) has been generated, the center pixel in the secondary band is multiplied by \( \gamma \) and the result is subtracted from the center pixel in the primary band. If the center pixel contains a potential target, the result will be a number much larger than zero. Conversely, for clutter, the output of the spectral filter should be near zero.

The spectral filters used in this research are two-band spectral filters. Each filter uses Band 2 as the primary band and Band 1 as the secondary band.
1.2.1 Two Band Ratio Filter

This filter requires two distinct processes. First, for every pixel in the original image data, the two band ratio filter computes \( z \), which is a ratio of the background energy in the primary band (Band 2) to the secondary band (Band 1). The energy ratio is defined as:

\[
x = \frac{\sum_{m=0}^{n-2} \sum_{n=0}^{n-2} P_2(x + m, y + n)}{\sum_{m=0}^{n-2} \sum_{n=0}^{n-2} P_1(x + m, y + n)}
\]

(6)

where \( P_2(x, y) \) is a pixel in the unfiltered Band 2 data and \( P_1(x, y) \) is a pixel in the unfiltered Band 1 data. Note that the point where \( m=n=0 \) is not included in these calculations.

Once \( z \) is calculated, the spectrally filtered value for the pixel, \( S(x, y) \), is computed as a function of \( B_1(x, y) \), \( B_2(x, y) \), and \( z \) as shown in equation 7.

\[
S(x, y) = B_2(x, y) - z \times B_1(x, y)
\]

(7)

\( B_1(x, y) \) and \( B_2(x, y) \) are the spatially filtered values for Bands 1 and 2, respectively.

1.2.2 Two Band Correlation Filter

This spectral filter algorithm uses a 5 x 5 pixel window around each pixel in two bands to generate a correlation factor, \( z \), which is an estimate of the correlation between the background data in the primary and secondary bands. After \( z \) has been generated, the value of the center pixel in the secondary band, \( B_1(x, y) \), is multiplied by \( z \) and the result is subtracted from the center pixel in the primary band \( B_2(x, y) \).

The value for alpha is defined as:

\[
x = \frac{\sum_{m=0}^{n-2} \sum_{n=0}^{n-2} P_2(x + m, y + n) \times P_1(x + m, y + n)}{\sum_{m=0}^{n-2} \sum_{n=0}^{n-2} P_1(x + m, y + n)}
\]

(8)

Once \( x \) is calculated, the spectrally filtered value for the pixel, \( S(x, y) \), is computed as a function of \( B_1(x, y) \), \( B_2(x, y) \), and \( z \) as shown in equation 9.

\[
S(x, y) = B_2(x, y) - x \times B_1(x, y)
\]

(9)

where \( B_1(x, y) \) and \( B_2(x, y) \) are the spatially filtered values for Bands 1 and 2, respectively.

1.3 Background Normalizer and Thresholder

This processing stage detects exceedances in spatially spectrally filtered data and outputs a list of exceedances to a tracking algorithm. The most common approach used for target detection in IR imagery is the contrast box algorithm which is a derivative of constant false alarm rate (CFAR) algorithms used in radar signal processing. Contrast box algorithms assume that the intensity in areas which contain targets will differ significantly from the intensity of their backgrounds.

In 1981, Burton and Benning evaluated four detection algorithms for IR imagery: the contrast box, the double gated filter, the spoke filter and the superslice algorithm [10]. The contrast box algorithm developed by Texas Instruments demonstrated the best performance.

Figure 3 shows the geometry for contrast box algorithms. Two windows or boxes slide across every pixel in the image. The inner box is matched to the size of the target and statistics are computed for the pixels within this box. This box is the target box. A second box is used to compute the background statistics around the center pixel and this box is generally much larger than the target box. The mean and standard deviation of the pixels within each box are then computed to create a contrast metric between the target and its background.

In Figure 3, \( \mu_t \) is the mean of the target box, \( \mu_b \) is the mean of the background box, \( \sigma_t \) is the standard deviation of the target box, and \( \sigma_b \) is the standard deviation of the background box.

![Figure 3. Geometry for the contrast box algorithm.](image-url)
where \( S(x, y) \) is the value of the spectrally filtered pixel. Note that the term where \( m = n = 0 \) is not included in the background mean calculation.

The contrast metric for each pixel is then compared to a threshold, \( K \), provided by the user. Pixels whose contrast metric exceeds the threshold are declared exceedances and their intensity values \( S(x, y) \) and their average background energy are output to the tracker.

2. IR Data Generation

The IR imagery used in the evaluation process consisted of simulated targets embedded in simulated scene data. The background data was created from an IRIM database and is a sequence of scenes imaged from an aircraft-based missile warning system. Simulated point targets with the spectral characteristics of missile threats were randomly inserted into the backgrounds.

Seven processes are used to create the realistic IR scene data used to evaluate the candidate IR missile warning algorithm suites. First, BLU MAX II is used to generate a flight path of an aircraft over the earth. The output of BLU MAX II is a time history of the aircraft, and hence the IR sensor's location with respect to the ground. This output is used as input to the Trajectory Analysis Program (TRAP) or the Enhanced Surface-to-Air Simulator (ESAMS).

TRAP is used to predict the performance of a launch aircraft, an air launched ground missile and a target in an air to air engagement scenario. TRAP uses the characteristics of the aircraft and the missile, including guidance and control system models, to produce an output containing the trajectory for the aircraft and missile relative to an earth fixed inertial frame of reference. The output is then converted to a sensor frame of reference.

ESAMS is similar to TRAP except that it provides modeling for surface-to-air engagements rather than air to air engagements. The output of ESAMS is a trajectory file similar to that generated from TRAP.

The next process in creating the data is calculating the IR signatures of the launch aircraft and the missile. The launch aircraft IR signatures are derived from the FID signatures database. The missile signatures, however, were derived from two sources: the FID signatures database and from the Signature of Air-to-Air Missiles After Burnout II (SAAMBO II) simulation program. The FID data base provides signatures for missiles during their burn stages, and SAAMBO II was used to model the post burnout missile signatures.

To create the background, the target aircraft's trajectory file is input to IR Background Simulation Computer Program (IRSBUS). This program creates the IR background as viewed from the target aircraft's sensor by projecting the sensor pixels onto the ground. The ground data is supplied from the IRIM Port Heunuene database. IRSIMAS inputs the trajectory file of the target aircraft and outputs a set of background scenes which have been attenuated for atmospheric transmission and optic effects.

The missile and launch vehicle signatures are attenuated for atmospheric and optic effects through the IR TARGET program. This program inputs the flight profiles of the target aircraft, the launch aircraft and the missile. The output from this program is a projection of the missile and launch vehicle signatures onto the FPA.

The last data generation step combines the background scene with the target scene by adding the pixel irradiances. The program AJ DATA also adds simulated electronic noise to the final image. The final output is a digitized IR scene of targets embedded in backgrounds with varying amounts of noise.

The data set used in this research consisted of 100 sequential frames of registered MWIR data. Each frame represents imagery from a 128x128 pixel InSb FPA that has been imaged at 30 frames per second. The inserted launch aircraft and missile targets were removed from the data set. Then, one hundred point targets with the same spectral characteristics as missiles in their burn stages were randomly inserted into each data frame resulting in 10,000 total targets. The six algorithm suites were then tested using the two data sets.

3. Data Collection Method

To measure the effectiveness of each algorithm suite, a false alarm rate versus missed detection rate metric is used. The signal to noise ratio for all targets in the data sequence was varied from 1.1 to 2.0 in steps of 0.1. All targets were initially inserted with a signal to noise ratio of 1.1 in the primary band, Band 2. Ten different values, \( K \), were used as the threshold of the background normalizer and thresholder and the number of missed detections and false alarms were recorded for each value of \( K \). The missed detection rate (MDR) is defined as the number of missed detections divided by the total number of inserted targets. The false alarm rate (FAR) is defined as the number of false alarms divided by the total number of pixels. The MDR and FAR recorded by varying \( K \) resulted in the data points needed for one performance curve for each input signal to noise ratio. Figure 4 shows an example family of curves for the performance of Algorithm Suite 4. As expected, the performance of Algorithm Suite 4 improves, in terms of fewer missed detections and false alarms, with increasing input signal to noise ratios.
4. Results and Conclusions

Figures 5, 6, 7, and 8 show the performance of each algorithm suite against target with signal to noise ratios of 1.1, 1.2, 1.3, and 1.5, respectively. Table 2 shows the relative computational complexity of each algorithm suite in terms of the number of adds, multiplies, logical operations and divides for each pixel processed. Note that the number of additions includes subtractions as well.

Figures 5, 6, 7, and 8 show that the performance of the algorithm suites varied little based on the spatial filter used. This result is not unexpected since the spatial filters used in this research were all designed to pass point targets.

The results also show that the ratio spectral filter performs as well as the correlation spectral filter when evaluated against registered imagery. The ratio spectral filter requires fewer computations, which may alleviate the processing requirements of IR missile warning systems. The performance curves show that good detection performance can be achieved against targets with low signal-to-noise ratios by using relatively simple two-dimensional IR missile warning algorithm suites.
Figure 7. Performance of Algorithm Suites Against 1.3 SNR Targets

Figure 8. Performance of Algorithm Suites Against 1.5 SNR Targets

Table 2. Per Pixel Computational Complexity of Algorithm Suites.

<table>
<thead>
<tr>
<th>Suite</th>
<th>Add</th>
<th>Multiply</th>
<th>Logical</th>
<th>Divide</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>107</td>
<td>2</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>207</td>
<td>50</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>99</td>
<td>1</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>99</td>
<td>49</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>99</td>
<td>1</td>
<td>28</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>99</td>
<td>40</td>
<td>28</td>
<td>2</td>
</tr>
</tbody>
</table>

References