Automatic target detection in forward-looking infrared imagery via probabilistic neural networks

Jesmin F. Khan,[1,*] Mohammad S. Alam,[2] and Sharif M. A. Bhuiyan[1]

[1]Department of Electrical and Computer Engineering, University of Alabama in Huntsville, Huntsville, Alabama 35899, USA

[2]Department of Electrical and Computer Engineering, University of South Alabama, Mobile, Alabama 36688, USA

*Corresponding author: khanj@eng.uah.edu

Received 30 May 2008; revised 28 November 2008; accepted 7 December 2008; posted 15 December 2008 (Doc. ID 96784); published 12 January 2009

This paper presents a technique for automatic detection of the targets in forward-looking infrared (FLIR) imagery. Mathematical morphology is applied for the preliminary selection of possible regions of interest (ROI). An efficient clutter rejecter module based on probabilistic neural network is proposed, which is trained by using both target and background features to ensure excellent classification performance by moving the ROI in several directions with respect to the center of the detected target patch. Experimental results using real-life FLIR imagery confirm the excellent performance of the detector and the effectiveness of the proposed clutter rejecter module. © 2009 Optical Society of America

OCIS codes: 040.2480, 100.4996.

1. Introduction

In several scientific as well as engineering disciplines, the challenges of automatic target recognition (ATR) have been researched [1]. The fundamental step of ATR is the automatic target detection (ATD) technique, which determines whether an unknown input image contains a desired object and, if it does, estimates the pixel location of that object in the image. The main tasks of ATR is to analyze an interesting candidate object identified by an ATD algorithm, determine whether that object is a target or a clutter, and finally assign a class label to that object.

The first stage of an ATD system is a preprocessing stage in which a target detector extracts possible regions of interest (ROI) containing potential targets. The second stage is the feature extraction, which should be done in such a way that the selected features are able to represent the target image effectively. By using a confidence measure, the third stage—a clutter-rejection stage—rejects the false targetlike objects (clutter) and retains true targets. This stage mainly classifies each candidate target image provided by the detector into clutter or desired target by using the features calculated at the second stage. In this paper, we focus on all three stages.

An ideal ATD system will exhibit the properties of a low false alarm rate (conferring a nontarget as a target) and miss rate (presenting a target as a nontarget) and, at the same time, obtain a high true positive rate (the detection and confirmation of a true target). The target detector operates on the whole image and usually finds the correct target as well as selects a number of background regions as potential targets (false alarms or clutter). Therefore a well-defined clutter rejecter must be devised and attached to reject most of the false alarms or clutter produced by the detector, and clutter-rejection techniques play a very important role in ATD, because overall ATD performance greatly depends on correct classification results.

We demonstrate an ATD algorithm in forward-looking infrared (FLIR) imagery based on nonlinear morphological operators, feature-extraction technique by means of a set of Daubechies low-pass wavelet

0003-6935/09/030464-13$15.00/0

filters, and clutter-rejection system based on probabilistic neural network (PNN). The combined ATD, effective feature extraction, and clutter-rejection algorithms maximize detection-classification performance on FLIR imagery and exhibit excellent detection performance for both hot and cold targets present in complex backgrounds. The techniques described in this paper for detection, clutter rejection, and classification are tested on FLIR image data.

The theme of this research is to perform an initial target detection at the first frame of a sequence so that it can be incorporated with the target-tracking algorithms, where it is assumed that the target location is known in the initial frame [2–6]. Therefore, in this paper, the detection and clutter-rejection tasks are implemented for only the first frame of a sequence. An overview of the proposed approach is presented in Fig. 1, which consists of the following steps:

1. Given a image sequence, several frames other than the test frames are randomly selected from that sequence as training images. Using ground truth data, the target region and a few background regions are located at the training images.
2. Wavelet filters are applied to the training images to extract feature vector for each pixel in the selected target and background regions. A proper label is marked to those feature vectors as either target or clutter. Then we formulate a clutter rejecter based on neural network, which is trained by utilizing those labeled feature vectors.
3. For detecting the target location in the test frame, an efficient morphological operation-based scheme is designed, which extracts all potential ROI that may contain the target. Here the proposed ROI-extraction technique does not use any target-specific information except the rough dimension of the target.
4. The wavelet filters are applied to the test image to extract the same features from all the extracted ROI. Then the trained network, i.e., the clutter rejecter, accepts the feature vectors from each ROI and exploits the target and background signature information reserved from the set of the training imagery to yield a decision of the ROI being either target or clutter.

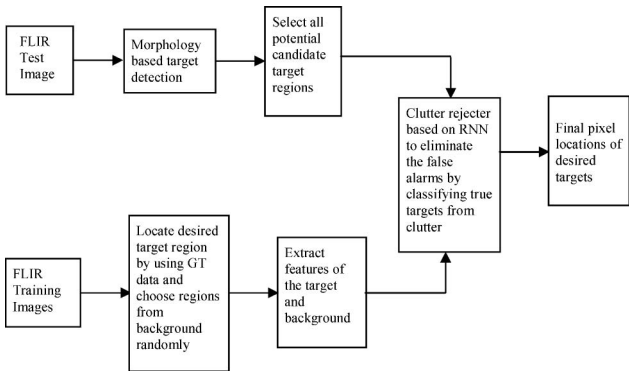


Fig. 1. Overview of the approach. GT, ground truth.

5. The shifting of the ROI is performed in different directions as a postprocessing step for the critical cases automatically, depending on the decision yielded by the PNN-based clutter rejecter. When the PNN shows that there is no target in the frame, the shifting process is initiated to ensure that the claim is not wrong. Thus the overall miss rate is reduced significantly in the proposed technique.

The rest of the paper is organized as follows. Section 2 presents a state-of-the-art on some existing ATD techniques. A brief description of the data set is given in Section 3. The morphology-based detection algorithm that operates on the entire image is explained in Section 4. Section 5 describes the computation of a set of texture filters based on wavelet transforms to extract the features of the target as well as the background and the training of the network using those features. The clutter rejecter that operates only on detected ROI is detailed in Section 6. In Section 7, an improvement of the presented classifier is proposed to handle a major problem in FLIR ATR, the poorly centered targets generated by the morphological preprocessing stage. Section 8 demonstrates the results of the algorithm on several different image sequences. The overall performance of the proposed technique and a comparison result with another FLIR ATD method are also presented in Section 8. Finally conclusions are drawn in Section 9.

2. Related Work

Over the past three decades, substantial research on ATD and ATR has been done. Some familiar methods for achieving this goal include feature-based classification exploiting filtering methods [7], statistical techniques [8], model-based techniques [9,10], and maximum likelihood algorithms [11]. The detection and discrimination of targets in infrared imagery is performed by applying a quadratic filtering method [7]. In [8–10], in general, the potential targets are segmented from the background, and boundary edges and internal features are measured and fitted to either statistical or structural models, whereas in [11], instead of performing separate steps of segmentation and feature extraction, the configuration of targets is directly estimated from the measured data. Target recognition techniques based on mathematical morphology are ideally suited for the analysis of shapes in a cluttered scene. Much work has been done with image analysis using mathematical morphology [12–14]. Crimmins and Brown [12] showed how mathematical morphology could be used to locate shapes by matching both the image and its complement. Shapiro *et al.* [13] and Shih and Mitchell [14] describe methods using mathematical morphology to extract primitives or parts of an object. Research effort in the past decade have demonstrated that morphological image processing offers potential for achieving significant improvements in subclutter visibility for unresolved targets in infrared imagery [15,16].

A number of target detection tasks in FLIR imagery based on neural network have already been proposed as well, both by the scientists and by the researchers working in academia and industry. A survey on neural-network technology for ATR is presented in [17], and an evaluation of neural-network-based ATR is performed in [18]. Some researchers approached through a set of intuitive image features to form a confidence image and a multiple-layer perceptron (MLP) network, where most of the detection efforts is expended in a few attractive regions highlighted by the preprocessing stage [19,20]. There are techniques of utilizing a neural-network-based target detector after the combination of images from different band [21,22] or fusing the detection results from several neural-network-based algorithms [23]. In [23], a modular neural-network classifier has been applied to the problem of ATR in FLIR imagery based on several independently trained neural networks, where each neural network makes a decision based on local features extracted from a specific portion of the target image. Finally, the classification decisions of the individual networks are combined to determine the final classification. ATD algorithms based on the feed-forward neural network have been reported in [24], where conventional image processing algorithms, such as hit/miss filtering, difference of Gaussian filtering, and region clustering in conjunction with a neural network (MLP) for classification are used to form a complete ATR system. In [25], a hybrid neural-network system is presented, called the pulse coupled neural net, for ATR, which is an iterative neural network in which a gray scale input image results in a one-dimensional time signal that is the input of a feed-forward pattern recognition net. Besides, an ATD algorithm based on the time-delay neural network [26] and other neural architectures [27–29] have been proposed as well.

3. Description of the Forward-Looking Infrared Image Data Set

Our FLIR image sequences supplied by Army Missile Command are acquired by an infrared camera installed in a moving platform, and the pedestal, on which the camera is lodged, is an aircraft. As the imaging system platform is not stable during the exposure time, all the objects in the image sequence are affected by the motion of the platform, and as a result, the recorded image would be blurred, and minute details of the desired image would be lost. Thus real-life FLIR imagery demonstrates a number of well-known challenges that have been previously well documented in [1,30,31] and make target detection in FLIR imagery a very challenging research topic [32].

For our experiment, we have 50 different sequences, where each sequence composes of a number of frames (130 to 779 frames per sequence), and each frame has 8 bit resolution of 128×128 pixels with one or more targets posed somewhere. In general, the image sequences are closing sequences, i.e., the target is at the farthest distance from the sensor and then gradually becomes closer to the observer as the later frames appear in the scene. Thus the target is rarely visible at the first frame because of being very small with no distinguishable target signature. The size and signature as well as the orientation of the target change from one frame to the next. For those sequences, ground truth data are available for all the targets at each frame, which includes the location of the targets and their height and width. In all those sequences together, there are a total of 103 targets of 10 different types, such as tank, truck, mantruck, M60, armored personnel carrier (APC), bradley, pickup, target, testvan, and van, that need to be detected.

4. Morphology-Based Target Detection

In this paper, a detection process based on the application of simple nonlinear gray-scale morphological operations has been utilized that leads to real-time implementations [33]. Morphological operations decompose the given input image into a filtered image. Here the proposed procedure for detection has several stages within morphological framework. The basic morphological operations of dilation, erosion, opening, and closing for gray-scale images are exploited to accomplish the different phases in the progression or development of our ROI extraction algorithm. In doing so, we have formulated an efficient and distinctive combination with proper ordering of those operators for extracting targets from FLIR imagery, where targets happen to appear as bright objects.

The two most basic operations in mathematical morphology are erosion and dilation [34]. The basic effects of the dilation and erosion operators on an image are to gradually enlarge the boundaries of regions of foreground pixels and to erode away the boundaries of regions of foreground pixels, respectively. Other than dilation and erosion, opening and closing are two important basic morphological operators [34], which are a combination of erosion and dilation and often used to select or suppress features of a certain shape, e.g., to remove noise from images or to select objects with a particular direction. Very simply, the opening and closing are defined as an erosion followed by a dilation and a dilation followed by an erosion, respectively, using the same structuring element (SE) for both operations.

Figure 2 shows a block diagram of the approach used to find the potential candidate target regions and the corresponding output images at different stages of the target detection module. In our technique, as shown in Fig. 2, dilation is used for edge detection by applying the dilation operator to an image and then subtracting away the original image from the dilated image. This process highlights the new pixels at the edges of objects that are added by the dilation. Based on the experiment on the available image sequences, it has been found that the images of all the previously mentioned ten different types of

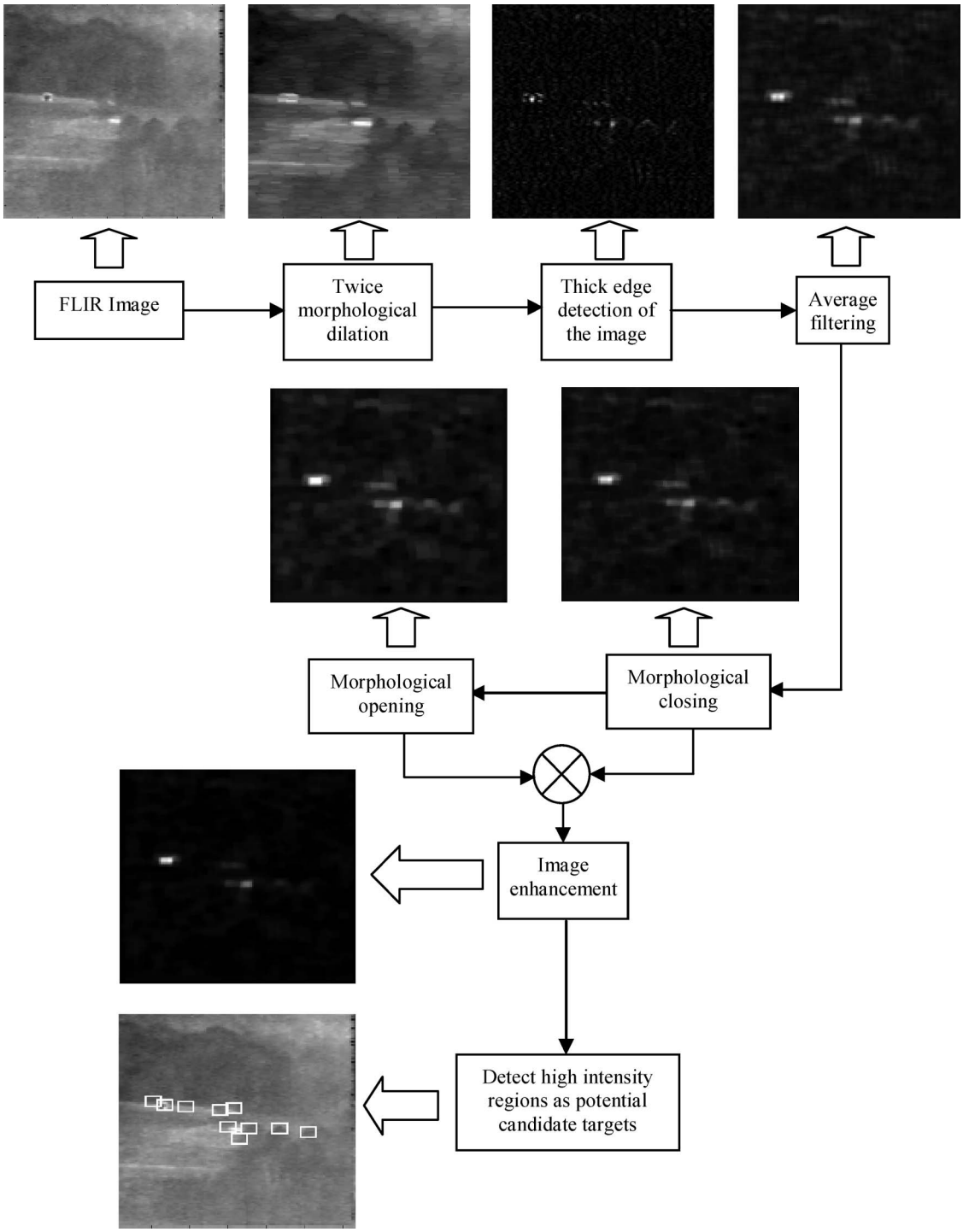


Fig. 2. Potential target detection module.

targets are usually wider than tall from all views. Therefore, in this work, the dilation is made directional by using unsymmetrical SEs, e.g., a SE that is three pixels wide and one pixel high (3 × 1) and that dilates in the horizontal direction only. Moreover, to provide additional reasoning for the use of this heuristic shape (3 × 1) for the SE, we have conducted experiments with two different shapes such as 1 × 3 and 3 × 3. We have also shown the measures for the hit rate and the false alarm rate in Table 1,

Table 1. Regions of Interest Detection Results Comparison for Three Different Shapes for the Structuring Element

Shape of the SE	Total Number of Sequences	Total Number of Targets	Number of Targets Marked as a ROI	Number of Misses
3 × 1	50	103	95	8
1 × 3	50	103	39	64
3 × 3	50	103	71	32

Table 2. Regions of Interest Detection Results Comparison for Different Number of Dilations

Number of Dilations	Total Number of Sequences	Total Number of Targets	Number of Targets Marked as a ROI	Number of Misses
1	50	103	76	27
2	50	103	95	8
3	50	103	93	10
4	50	103	82	21

which is generated by applying the method on 50 different sequences with 103 total targets posed somewhere in the first frame.

In FLIR images, the object is often so small that it may only be a few pixels in size and have a low contrast with its background. In our detection algorithm, to overcome the difficulty of detecting small and low contrast objects, morphological dilation with a centered horizontal element, s, of width 3 is applied twice so that little objects become more perceptible. Then the subtraction of the twice-dilated image from the original image provides broad edges. There has been a significant improvement in the detection performance from the employment of the dilation twice in comparison with only once. The best choice for the number of successive dilations has been determined experimentally using the collection of the 50 sequences. The experimental determination of number of dilations that maximizes the true ROI detection rate is shown in Table 2. As this table shows, the highest ROI detection rate is obtained when the number of dilations is two. After that, the number of targets selected as a ROI decreases, and hence the number of misses increases.

Average filtering, a region-filling process, is applied next to join the detached nearby edges of an object in conjunction with filling the holes encompassed by that linked edge, where the averaging filter is of size $w \times h$, which is the expected size of the target. Thereafter, morphological closing and opening with a $w \times h$ rectangular SE are performed to ensure the least overall distortion of the original image and, at the same time, to realize the attractive attributes offered by those operations. Closing tends to fuse narrow breaks and long thin gulfs, eliminate small holes, fill the gaps in the contour, and as opposed to closing opening breaks narrow isthmuses, eliminates thin protrusion [34]. In our application, the closing operation removes dark details and brightens the filtered image by initial dilation and darkens the image by the subsequent erosion without reintroducing the details removed by the dilation. The opening operation is employed to remove small light details while leaving the overall gray levels and larger bright features relatively undisturbed [34]. The initial erosion of opening operation not only removes the small details, but also darkens the image. On the other hand, the subsequent dilation of opening operation again increases the overall intensity of the image without reintroducing the details totally removed by the erosion.

As the final intensity enhancement process, the two images obtained from the closing and opening operations are multiplied to result in the final morphologically preprocessed image, F. The final preprocessed image is used to detect candidate objects, i.e., the ROI, by finding the local maxima. To determine the ROI from F, the pixel location associated with the maximum intensity value is first chosen and designated as the first local maximum. Then a region is grown around the local maximum with a number of pixels equal to the expected size of the target. The gray-level values of all the pixels situated within the region, i.e., the neighborhood around the local maximum (peak pixel), are set to zero, so the subsequent detections will not choose a pixel location corresponding to the same candidate target region. The output images at different stages of the target detection module are given in Fig. 2. The search for valid candidate target regions starts at the local maxima with the highest gray-level value and gradually descends and is repeated until the number of valid candidate target regions is greater than a predefined number of detections, which is chosen to be five for initial neural-network-based clutter-rejection module. But if the network gives decision for all the five detected regions as clutter and the shifting process is initiated, then this number is made larger, such that if the clutter rejecter fails for one target, then predefined number of detections is made to be ten, and for the failure of two targets or more, this number is 15. Following this procedure, there are a total of 335 extracted ROI from the 50 sequences to detect the 103 targets, among which the 95 ROI contain a target. In Table 3, the result has been reported on what percentage of ROI actually contains a target after the completion of algorithm's first stage. Moreover, a quantitative detection result, for example, the hit rate, i.e., the total number of extracted ROI that contain the targets after the first stage of the ROI detection algorithm, is given in Table 3.

From the analysis of all the image sequences, we find that the presented morphological-operator-based ROI extractor performs excellent for detecting possible target regions from the test image without any prior knowledge about the target. Being

Table 3. Regions of Interest Detection Results

Total Number of Sequences	Total Number of Targets	Number of Targets Marked as a ROI	Number of Misses	Total Number of ROI Extracted	% of ROI that Contain a Target
50	103	95	8	375	25.33%

extremely small and not having enough signature of the target in the formulated morphological framework, the application of the dilation operator twice enhances the determination of the edges. Moreover, the overall combination and ordering of those operators has been found to be very successful for extracting targets from FLIR imagery where targets happen to appear as bright objects.

5. Feature Extraction

To extract image features and to enhance those features with respect to their surroundings, wavelet-based processing algorithms are superior due to their abilities to discriminate different frequencies and to preserve signal details at different resolutions [35]. In this paper, we exploit useful properties of wavelet filters [36–38] to provide a robust method for object detection, even when the target is very small in size and not apparent to human eye in the image sequence due to its low contrast. For our purposes of feature extraction, the image is described in terms of wavelet decomposition since wavelets encapsulate information on discontinuities in a very succinct way [39]. Wavelets are mathematical functions, and each wavelet is associated with a scaling function, $\phi(x)$, via two-scale equations: $\phi(x) = \sum_{n\in\aleph} h_n\phi(2x-n)$ and $\psi(x) = \sum_{n\in\aleph} g_n\phi(2x-n)$, where $\aleph \equiv$ the set of real integers.

Each wavelet can be characterized by a finite set of coefficients, h_n and g_n, which constitute low-pass and high-pass filters, respectively [36]. The advantage of using these wavelet filters is that they are robust to noise.

Our purpose in this work is to design a wavelet-based feature extraction algorithm that measures texture quality along the most perceptual dimensions. The approach we have adopted uses a series of Daubechies wavelet filters h_n of lengths 4, 6, 8, and 10. The objective of the feature extraction is to compute the horizontal and vertical directionality, where texture directionality along the x axis is based on the horizontal convolution with the coefficients of four wavelet bases, and similarly, the directionality function along the y axis is based on the vertical convolution with those wavelet coefficients. The directionality along the x axis represents the vertical edges, while directionality along the y axis represents the horizontal edges. The horizontal and vertical convolution with the four wavelet bases generates a set of eight texture features [40,41].

6. Probabilistic Neural Network Clutter Rejecter

The function of a clutter rejecter is to further examine the locations indicated by the ROI yielded by the morphology-based detector. Neural network is the most versatile tool for the classification task, where, for most cases, only the samples of the desired target images are used for the training purposes. But only target-sample-based training is not adequate for the present case, because (1) the target is almost blended with the background, (2) there is only a limited diversity of the target in the training set, (3) the target consists of few pixels, (4) there is a similarity of considerable extent between target and background, (5) targetlike objects present in the input scene, and (6) contrast difference is very low between the target and the background. To get rid of those intricacies in target classification, both target samples and background samples are exploited for training purposes in this work. It is found that training only with the target, in the case of our image sequences, results in sharp difference between training and testing performance. In this paper, PNN, one of the two variants of radial-basis neural network (RNN), is used to reject clutter and accept targets. The proposed PNN-based clutter-rejection system given in Fig. 3 has the following stages:

1. For detecting a target in the first frame of an image sequence, we chose 20 frames from another sequence in which the same type of target is present. To detect the target, which is present only in a single image sequence consisting of N frames, we chose 20 frames from the same sequence. In that case, we excluded the first 10 frames to be considered as a part of the training set and randomly chose 20 frames for training purposes from the remaining $(N-10)$ frames.
2. Using the ground truth data, we extracted the target patch and five randomly chosen background patches from each training image to generate the set of training data. We convolved all the training images with the four wavelet filters horizontally and vertically to yield eight feature images from each training image.
3. For each of the 20 training images, there are eight feature images, and the selected target and background patches from those 960 images give the wavelet-based texture features for target region as well as for background. The PNN is then trained by those features so that it can discriminate true target later based on the extracted features from the detected regions in the test image.
4. As described in Section 4, given the test frame, the ROI identification module based on morphological operation selects some predefined possible candidate target patches at the highest intensity values of the final preprocessed image. A rectangular window was centered on each selected ROI based on the tentative size information of the target obtained from the training images, and any other windows at the same place of previously marked windows were

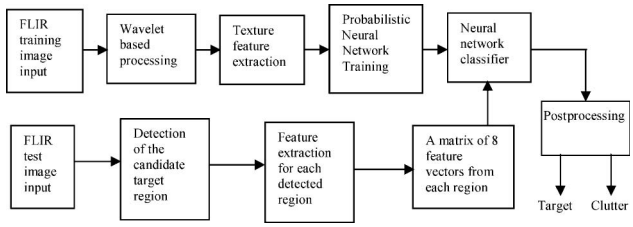


Fig. 3. PNN-based target classification system.

ignored. The test frame was then convolved with the same wavelet filters for extracting features from each ROI.

5. Finally, the clutter-rejection module deals with each ROI separately for extracting features and returns a classification matrix of eight feature vectors for each ROI. As the network has already been trained, it is ready to make classification decision. The network accepts the feature matrix yielded from each ROI as an input and produces an output row vector corresponding to that ROI. The desired output is chosen to be a row vector of eight 1s for proper target class and a row vector of eight 10s for nontarget or clutter class. But for some ROI, the output row vector contains less than eight 1s with a 10 for the remaining elements or less than eight 10s with a 1 for the remaining elements of the output row vector. In that case, the output of the network for that ROI is interpreted as the probability of being in either target class or clutter class, depending on the number of 1s or 10s in output. If there is no 1 in any of the row vectors of the network output, then it means that there is no target in that test frame, which may be either a correct decision or a miss.

The details of the proposed PNN clutter-rejection scheme are further discussed in the following subsections.

A. Preparation of Training Data

The following three steps should be carried out for classifying target:

- Describe the target and the background using a model.
- Optimize both models using training data.
- Find true targets by examining each ROI to check if it is accurately described by the models.

The first step, therefore, is to generate a model of the target and the background by computing $n = 8$ features for each pixel in the known target region and also in the randomly chosen background regions in a training image. The feature-extraction module uses information from the eight filtered images for each image in the training set. For every training image from the target and background regions, we obtain feature matrices $\mathbf{f}_t$ and $\mathbf{f}_b$, respectively:

$$\mathbf{f}_t = [f_t^1 f_t^2 f_t^3 \ldots f_t^8],$$

$$\mathbf{f}_b = [f_{b1}^1 f_{b1}^2 \ldots f_{b1}^8 f_{b2}^1 f_{b2}^2 \ldots f_{b2}^8 \ldots\ldots f_{b5}^1 f_{b5}^2 \ldots f_{b5}^8].$$

$f_t^1, f_t^2, \ldots, f_t^8$ are the vectorized images of the eight filtered target patches, and $f_{bi}^1, f_{bi}^2, \ldots, f_{bi}^8$ are the eight filtered vectorized images of the ith background patch.

Each feature vector of target and background can be defined as $f_t^i = [x_1^i, x_2^i, x_3^i, \ldots, x_p^i]$ and $f_{bk}^i = [y_{k1}^i, y_{k2}^i, y_{k3}^i, \ldots, y_{kp}^i]$, where $i = 1, 2, 3\ldots, 8$; $k = 1, 2, 3\ldots, 5$; p is the number of pixels in the target or background patch such as $p = w \times h$, where w and h are the width and height of the target window; x_j^i is the target pixel value in the ith filtered image; y_{kj}^i is the background pixel value of the kth randomly chosen background patch in the ith filtered image; and $j = 1, 2, 3\ldots, p$.

Thus, for each of the training images, there are 8 feature vectors from the target and 40 from the background, i.e., in total there are 48 feature vectors. Since in our application we have selected 20 images as our training set, the feature training set contains 960 training feature vectors ($8 \times 20 = 160$ from the target area and $5 \times 8 \times 20 = 800$ from the background area). These feature vectors are collected into a matrix, $\mathbf{X} = [\mathbf{f}_t\, \mathbf{f}_b]$, and each feature vector is labeled with an appropriate class. The classes are as follows:

1. Desired target.
2. False target, i.e., clutter.

B. Training the Network

Neural networks have been used with great success in the past on target recognition problems [17–29]. Neural networks can be trained by data with appropriate class labels (each output represents one class) and then used directly as a classifier of new data. Back-propagated delta rule networks (BPNs) (sometimes known as MLPs) and RNNs are both well-known developments of the delta rule for single layer networks (itself a development of the perceptron learning rule). Both can learn arbitrary mappings or classifications. Further, the inputs (and outputs) can have real values. Like BPNs, RNNs can learn arbitrary mappings, and the primary difference between BPNs and RNNs is in the hidden layer. RNN hidden layer units have a receptive field with a center, which is a particular input value at which they have a maximal output. Their output tails off as the input moves away from this point. Generally, the hidden unit function is a Gaussian RNN and is trained by

- deciding on how many hidden units there should be and
- deciding on their centers and the sharpness (standard deviation) of their Gaussians.

RNNs have the advantages that they are more robust in dealing with poor training sets than the MLPs, and one can add extra units with centers near parts of the input, which are difficult to classify.

As mentioned before, the neural network chosen for this research is a PNN, which is one kind of RNN, suitable for classification problems. Its design is straightforward and does not depend on training [42]. A PNN is guaranteed to converge to a Bayesian classifier, provided it is given enough training data. These networks generalize well, and the training scheme is very simple and fast [43]. As shown in Fig. 4, PNN consists of three feed-forward layers: input layer, pattern layer, and summation layer. Suppose there are Q input vectors, each of which is

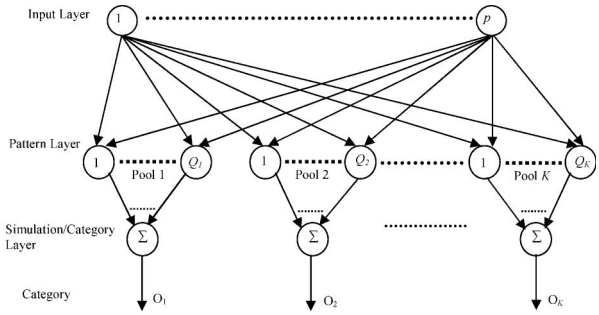


Fig. 4. PNN consisting of p input units, Q pattern units, and K category units.

p dimensional and sampled from K categories. The PNN for this case consists of p input units comprising the input layer, where each unit is connected to each of the Q pattern units; each pattern unit is, in turn, connected to one and only one of the K summation/category units. The connections from the input to pattern units represent modifiable weights, **IW**, which will be trained. Each category unit computes the sum of the pattern units connected to it.

The network architecture is given in Fig. 5. It is assumed that there are Q input vector/target vector pairs. Each target vector has K elements such that one of these elements is 1 in the row associated with that particular class of input and 0 elsewhere. Thus each input vector is associated with one of K classes. The first-layer input weights, **IW**, are set to the transpose of the matrix formed from the Q input vectors, $\mathbf{X^t}$. In Fig. 5, the input to the radbas transfer function is the vector distance between its weight vector ${}_j\mathrm{IW}$ and the input vector **X** multiplied by the bias b_j^1. The ||dist|| box in Fig. 5 accepts the input vector **X** and the single row input weight matrix and produces the dot product of the two. The transfer function for a radial basis neuron is $\mathrm{radbas}(n) = e^{-n^2}$. The second-layer weights, **LW**, are set to the matrix **T** of target vectors. The multiplication $\mathbf{Ta^1}$ sums the elements of $\mathbf{a^1}$ due to each of the K input classes. Finally, the second-layer transfer function, *compete*, produces a 1 corresponding to the largest element of $\mathbf{n^2}$ and 0 elsewhere. Thus the network classifies the input vector into a specific K class because that class has the maximum probability of being correct.

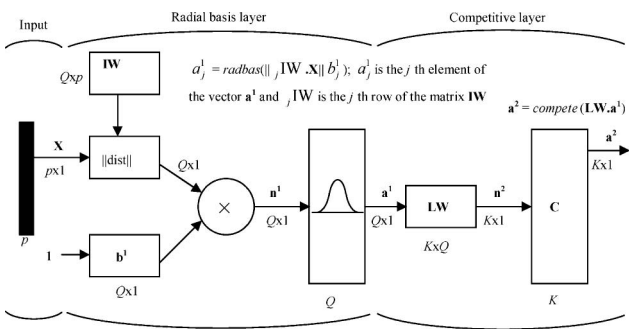


Fig. 5. Network architecture.

The input layer accepts the feature vectors and supplies them to all the neurons in the pattern layer. When a test input is presented, the ||dist|| box of this layer computes distances from the input vector to the training input vectors and produces a vector whose elements indicate how close the input is to the vectors of the training set. These elements are multiplied, element by element, by the bias and sent to the radbas transfer function. An input vector close to a training vector is represented by a number close to 1 in the output vector $\mathbf{a^1}$. If an input is close to several training vectors of a single class, it is represented by several elements of $\mathbf{a^1}$ that are close to 1. The pattern layer operates competitively, where only the highest match to an input vector wins and generates an output. The second-layer transfer function, *compete*, produces a 1 corresponding to the largest element of $\mathbf{n^2}$ and 0 elsewhere. Thus the network classifies the input vector into a specific K class, because that class has the maximum probability of being correct. In this way, only one classification category is generated for any given input vector. If the input does not relate well to any patterns programmed into the pattern layer, no output is generated [44]. Finally, the summation layer forms the weighted sum of the outputs from the pool in the pattern layer, and a *compete* transfer function on the output of this layer picks the maximum of these probabilities and generally produces a 1 for the desired class and a 0 for the other classes.

Wavelet filters are exploited to extract salient features from both the target and the background data of the training images. The feature vectors extracted for each pixel of the target and the background region are then labeled as either target or nontarget, depending on whether the original pixel belongs to the target or background area. Finally, the labeled target and background feature vectors are combined together to form the matrix **X** and fed into the PNN for training purposes. In our work the PNN is created in MATLAB using the command $\mathrm{net} = \mathrm{newpnn}(\mathbf{X}, \mathbf{T})$, where **X** has 960 vectorized feature images, each of which has p elements.

In this work, during training, a 1 is chosen as a class label for the desired target and a 10 for anything else in the matrix **T**. Unlike the feed-forward back-propagation network (FFBN) or generalized regression neural network (GRNN), PNN is not able to generate any integer or fractional number other than the selected two numbers (in our case, 1 and 10) as output, either with training data or test data. So there is no need to apply any threshold to the network output of a PNN, as it is necessary in case of FFBNs, GRNNs, and most of the other neural networks. Like all other networks, a PNN also requires training imagery of each target, and it can be retrained rapidly with a new target.

C. Clutter Rejection

In our work, the trained PNN serves as clutter rejecter. As shown in Fig. 6, the clutter rejecter needs to

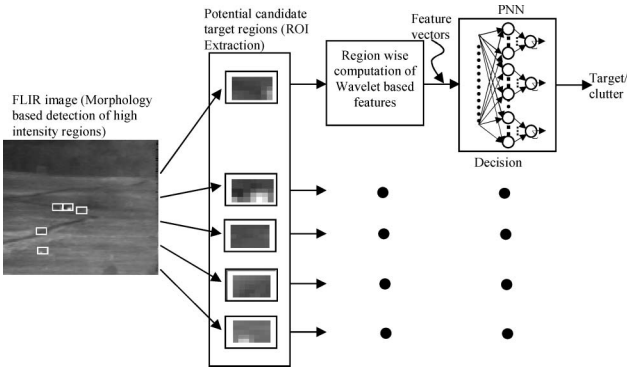


Fig. 6. Typical clutter-rejection module.

operate only on the small subset of the image (i.e., the potential target images or the ROI) indicated by the detector. Though the morphological detector does not assume any target-specific information as known other than the approximate size of the target, the clutter rejecter incorporates *a priori* known signature and texture information to discriminate between targets and backgrounds.

To discriminate true targets in the test frames, the same set of eight wavelet-based texture features is generated for every pixel in all the detected candidate regions in the image. Thereby the eight vectorized images for the eight features from each detected region form eight feature vectors, which are collected in a matrix. In this way, each detected region results in a matrix containing eight column vectors, where the size of each vector is $p \times 1$, and p is the number of pixels in each ROI. Those matrices are fed into the trained network for the classification. The identification module, which is based on probabilistic neural network, uses a supervised training set to develop distribution functions within a pattern layer. These functions, in the recall mode, are used to estimate the likelihood of an input feature vector being part of a learned category or class to provide an output for each detected region whether it belongs to the same distribution as the background or a target. Thus the decision about a potential candidate target is made either as a clutter or a real target based on the similarity between the extracted feature vectors from the training images and the corresponding representative feature vectors from a detected region.

7. Enhanced Clutter-Rejection System

A major problem in the previously described FLIR clutter-rejection system is the poorly centered targets generated by the morphology-based ROI extraction stage. The potential target regions generated by this preprocessing stage are generally off centered, and some images have only a part of the target. It is found that the off-centered potential target images generated by the preprocessing stage are the main cause of a substantially large number of false alarms resulting in the clutter-rejection stage. This ultimately results in rather poor performance of an ATD system on real-life data, though the same clutter-rejection system otherwise performs very well on the development set, in which targets are centered with the use of ground truth information. To overcome the problem of poorly centered targets, our clutter-rejection system is further improved by using the concept of shifted ROI. This step is triggered automatically, depending on the decision yielded by the PNN-based clutter rejecter based on the unshifted original ROI. When the PNN shows that there is no target in the frame, i.e., if the network fails to make a decision for any of the detected regions as a true target, in other words, if the network gives decision for all the detected regions as clutter, the shifting process is initiated to make sure the claim is not wrong. Thus we reduce the overall miss rate of the presented ATD technique. By this method, each detected ROI is moved in several directions, with respect to the center of the potential target image, to extract a number of shifted ROI for each region. Wavelet filtering is applied on all the shifted regions to extract feature vectors in the same way as described in Subsection 6.C for the case of original unshifted ROI and fed into the network for classification.

In this provisional mode of clutter rejection, the original ROI extracted by the detector as potential target chips are sent to an appropriate module for further processing. In this module, each original ROI is displaced in horizontal, vertical, and diagonal directions to extract several new ROI, which are then sent to the clutter-rejection module. The new shifted ROI corresponding to each original ROI allow us to extract useful features for the case of the off-centered target chips. Figure 7 shows how a representative ROI is displaced to have several new shifted ROI in the proposed method.

In the movement of the representative ROI, we use 1 original region and 24 displaced ROI for the implementation of our modified clutter rejecter. The 24 shifted ROI are extracted by moving the representative ROI in horizontal, vertical, and diagonal directions by one and two pixels with respect to the

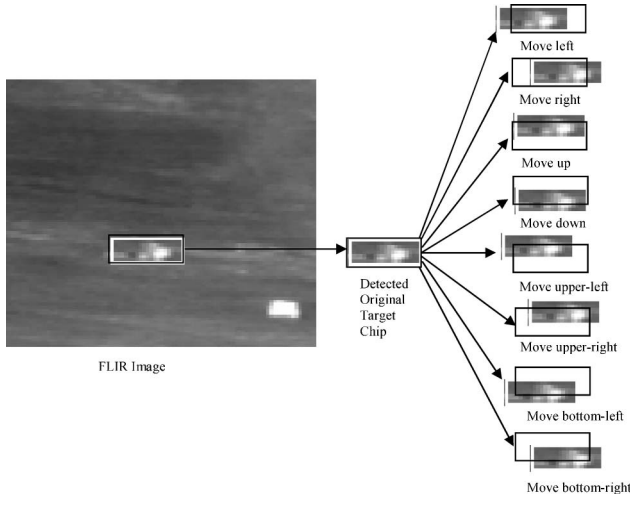


Fig. 7. Shifted ROI extraction.

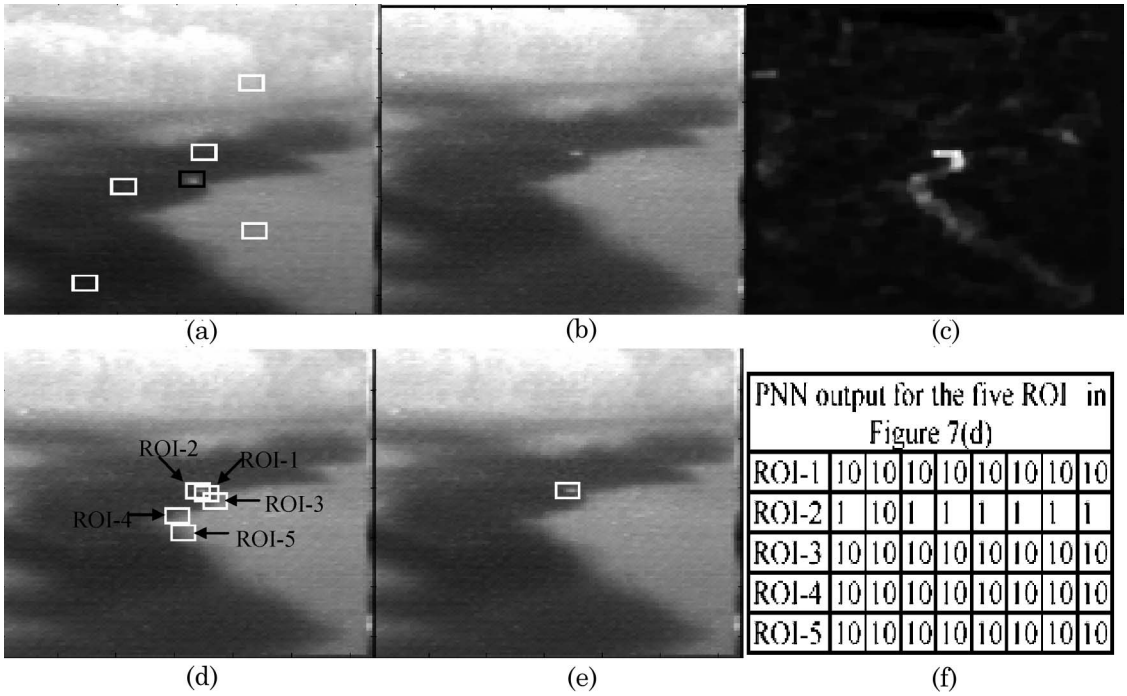
center of the potential target chip. After drawing out the shifted ROI, wavelet-filter-based texture feature extraction is performed in spatial domain. Like before, each shifted region generates eight feature vectors, which are arranged to form a matrix. This matrix of feature vectors is supplied to the PNN clutter rejecter to be categorized as either clutter or real target. It is not unusual that there may be one or more shifted ROI extracted from the same original ROI (which contains the off-centered target), which would appear as true target simultaneously with equal degree of probability. For such cases, fusion of the classification decisions from several shifted ROI is required to result in the final decision about the pixel location of the target in the input scene but not about the identity of the target in the original ROI. However, fusing the decisions, for that reason, is not crucial, because the applied shifting in any direction is not more than two pixels from the original location.

8. Experimental Results

In the experiments, there are 20 target and 100 clutter patches extracted for each sequence from the randomly selected 20 training frames utilizing the method discussed in Section 6, which are used for PNN training. The proposed system is applied to all the sequences of the given FLIR image database first with no shifting of ROI. For some cases, it is found that the potential target regions generated by the preprocessing stage are not well centered, and the performance of the clutter-rejection system deteriorates accordingly. However, the application of ROI shifting to these cases can successfully make a correct decision. Here results are shown for some of these sequences comparing the performance with and without shifting the ROI.

For illustration purposes, the analysis on the sequences L1720, M1410, L1813, L1817S1, and L2208 are presented by applying the proposed method of ATD. The available ground truth data for the test frame are exploited to evaluate the final detection result obtained from the proposed algorithm. There are total 735 frames containing the target M60 in the first sequence (L1720). To detect that target in the test frame of this sequence, we take 20 training frames for wavelet-based texture feature extraction.

Figure 8(a) shows a sample training frame of sequence L1720, where the black window indicates the target patch, and five randomly placed white windows indicate the clutter/background patches, which are utilized for PNN training. Figure 8(b) shows the test frame (the first frame) from where five ROI as possible target regions are selected by using the morphological target detection algorithm without any prior knowledge. Figure 8(c) shows the image output after the morphological image preprocessing. Figure 8(d) shows the detected possible candidate target regions, where the second ROI is well centered with the target, and the first ROI has a part of the target. The clutter rejecter/network accepts the feature vectors separately for each ROI and provides a decision either as true target or clutter. The final detection results after employing the clutter rejecter without ROI shifting is displayed in Fig. 8(e). Figure 8(f) presents the neural-network output for



PNN output for the five ROI in Figure 7(d)

ROI-1	10	10	10	10	10	10	10	10
ROI-2	1	10	1	1	1	1	1	1
ROI-3	10	10	10	10	10	10	10	10
ROI-4	10	10	10	10	10	10	10	10
ROI-5	10	10	10	10	10	10	10	10

Fig. 8. Sequence L1720. (a) One sample training image, where the black window contains the target, and the white windows contain the background; (b) test image; (c) morphologically processed image; (d) all the ROI selected by the detection module; (e) final detection result after the application of clutter-rejection module; and (f) PNN output.

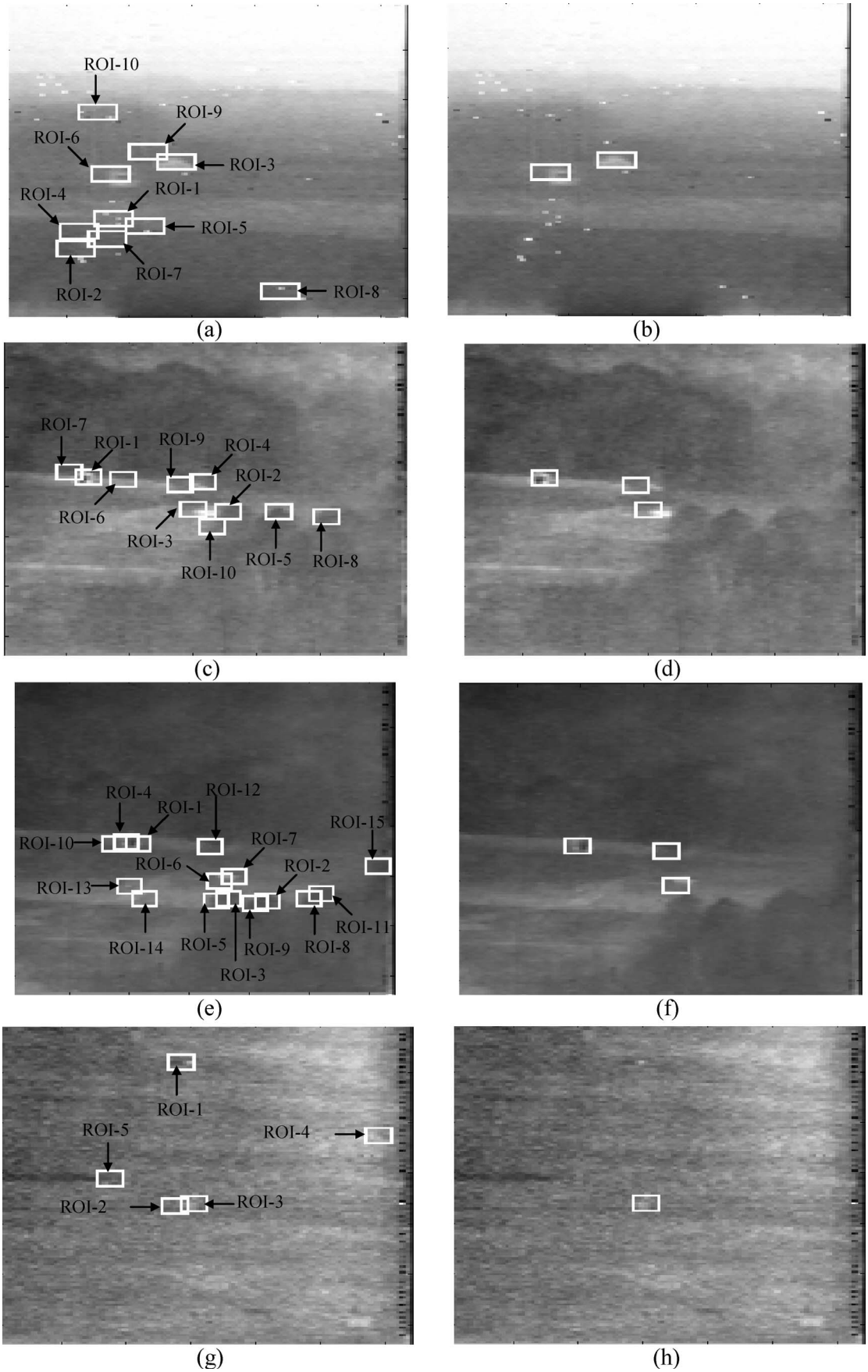


Fig. 9. Sequence M1410: (a) ROI and (b) final detection result. Sequence L1813: (c) ROI and (d) final detection result. Sequence L1815S1: (e) ROI and (f) final detection result. Sequence L2208: (g) ROI and (h) final detection result.

Table 4. Detection Results Comparison between the Presented Technique and the Algorithm in [33]

Algorithm	Total Number of Sequences	Total Number of Targets	Number of Targets Detected	Number of False Detection	Number of Misses
Proposed Method	50	103	84	1	18
Method of [33]	50	103	51	29	23

all the five preliminary detected regions, where the second region (ROI-2) shows up as target (M60), while the other regions show up as clutter or background. For each ROI there are eight feature images, and therefore the PNN generates a 1×8 output vector for each candidate ROI. For any ROI, if all the eight feature vectors match to the target feature vectors of the training set, the PNN output corresponding to that ROI will contain eight 1s. Similarly, for the matches with the background feature vectors of the training set, the PNN output will have 10s. For example, as shown in Fig. 8(f), ROI-1 matches with the background for all the eight features, and ROI-2 agrees with the target for all the features except the second feature. At the final step, if center location of the detected ROI-2 is within two pixels difference from the target's true center location provided in the ground truth data, the detection result is counted as a successful estimation. The detection results of four other sequences are demonstrated in Fig. 9.

We have compared our detection result with another ATD technique (presented in [33]), and the comparison is given in Table 4. The overall performance of our technique has also been reported in this table, where the method is applied on 50 different sequences with one or more targets posed somewhere in the first frame. In those sequences, there are a total of 103 targets of 10 different types, such as tank, truck, mantruck, M60, APC, bradley, pickup, target, testvan and van, that need to be detected. From this table, it is evident that the proposed scheme yields significantly better result than the method presented in [33]. In addition to the detection performance, the complexity, in other words, the computational costs of our algorithm, and the technique proposed in [33] are reported in Table 5. All the processing times given in Table 5 are recorded based on the simulation of MATLAB codes on a 2.59 GHz Pentium IV machine. Tables 4 and 5 prove the efficacy and potentiality of the presented technique that can be combined with any target-tracking algorithms for effectively performing an initial target detection at the first frame prior to the initiation of the target-tracking algorithm.

Table 5. Computational Cost Comparison between the Presented Technique and the Algorithm in [33]

Algorithm	Average Off-Line Training Time	Average Detection Time
Proposed Method	15 s	1.2 s
Method of [33]	13 s	1 s

9. Conclusion

The results presented in Section 8 demonstrate that the proposed algorithm has a great potential for ATD/recognition. The detection algorithm employs simple but constructive morphology-based nonlinear gray-scale operations for ROI extraction, which is thereby easily executable or implemental in real time, and computationally is fast and efficient. Furthermore, this algorithm is found to be robust for its ability to mark a target as a ROI under almost any condition, regardless of whether the target is hot or cold, small or large, present in highly cluttered or clear background, distinct or blended with surroundings, or in a conventional or distorted form. Followed by the preliminary detection, the clutter-rejection segment, which deals only with the ROI denoted as potential candidate targets, can decide unerringly whether a ROI is a target or clutter and then classify from other targets present in the same input scene. The PNN, i.e., the clutter-rejection module, draws conclusion based on the likeness between the extracted features from the known training images of the desired target and that from the unknown ROI. But when a ROI placed by the detector is off center with respect to the center of the target, the network exhibits that ROI as clutter/background, although it contains the whole target or a part of it. In such cases, all the preselected ROI are passed through the enhanced version of the clutter rejecter for further processing, where each ROI is displaced in all possible directions to extract several shifted ROI. After having features from those shifted ROI, a decision fusion needs to be applied due to the reason that more than one shifted ROI, extracted from the same original ROI, may appear as true target by the PNN. Hence, with the application of this modified version only when required, the overall performance of clutter rejection and classification is significantly improved. This presented shifting process can be utilized as a useful tool for any detection, recognition, classification, and/or matching problem. Though in this paper, according to our objective, only an initial detection is performed in a single frame of a sequence, it could be applied for any number of frames following the same procedure, which is obvious from the fact that the test frame (here the initial frame) is never included in the training data set.

References

1. B. Bhanu, "Automatic target recognition: state of the art survey," IEEE Trans. Aerosp. Electron. Syst. **AES-22**, 364–379 (1986).

2. A. Mahalanobis, "Correlation filters for object tracking target re-acquisition and smart aimpoint selection," Proc. SPIE **3073**, 25–32 (1997).
3. A. Bal and M. S. Alam, "Dynamic target tracking using fringe-adjusted joint transform correlation and template matching," Appl. Opt. **43**, 4874–4881 (2004).
4. M. S. Alam and A. Bal, "Improved multiple target tracking via global motion compensation and optoelectronic correlation," IEEE Trans. Ind. Electron. **54**, 522–529 (2007).
5. A. Dawoud, M. S. Alam, A. Bal, and C. Loo, "Target tracking in infrared imagery using weighted composite reference function-based decision fusion," IEEE Trans. Image Process. **15**, 404–410 (2006).
6. A. Bal and M. S. Alam, "Automatic target tracking in FLIR image sequences using intensity variation function and template modeling," IEEE Trans. Instrum. Meas. **54**, 1846–1852 (2005).
7. S. R. F. Sims and A. Mahalanobis, "Performance evaluation of quadratic correlation filters for target detection and discrimination in infrared imagery," Opt. Eng. **43**, 1705–1711 (2004).
8. D. Casasent and R. Shenoy, "Feature space trajectory for distorted object classification and pose estimation in SAR," Opt. Eng. **36**, 2719–2728 (1997).
9. B. Bhanu and J. Ahn, "A system for model-based recognition of articulated objects," in *Proceedings of the International Conference on Pattern Recognition* (IEEE, 1998), Vol. 2, pp. 1812–1815.
10. S. Z. Der, Q. Zheng, R. Chellappa, B. Redman, and H. Mahmoud, "View based recognition of military vehicles in LADAR imagery using CAD model matching," in *Image Recognition and Classification, Algorithms, Systems and Applications*, B. Javidi, ed. (Marcel Dekker, 2002), pp. 151–187.
11. A. D. Lanterman, M. I. Miller, and D. L. Snyder, "Automatic target recognition via the simulation of infrared scenes," in *Proceedings of the 6th Annual Ground Target Modeling and Validation Conference* (Keweenaw Research Center, Michigan Tech. University, 1995), pp. 195–204.
12. T. R. Crimmins and W. M. Brown, "Image algebra and automatic shape recognition," IEEE Trans. Aerosp. Electron. Syst. **AES-21**, 60–69 (1985).
13. L. G. Shapiro, R. S. MacDonald, and S. R. Sternberg, "Shape recognition with mathematical morphology," in *Proc. 8th Int. Conference on Pattern Recognition*, Paris, France, 27–31 October 1986.
14. F. Y. Shih and O. R. Mitchell, "Automated fast recognition and location of arbitrarily shaped objects by image morphology," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 1988), pp. 774–779.
15. V. Tom and T. Joo, "Morphological detection for scanning IRST sensor," Final Report TR-1167-90-1 (Atlantic Aerospace Electronics Corporation, 1990).
16. V. Tom and T. Joo, "Morphological-based front-end processing for IR-based ATR systems," Final Report (Atlantic Aerospace Electronics Corporation, 1992).
17. M. W. Roth, "Survey of neural network technology for automatic target recognition," IEEE Trans. Neural Netw. **1**, 28–43 (1990).
18. K. W. Przytula and D. Thompson, "Evaluation of neural networks for automatic target recognition," in *Proc. IEEE Conf. Aerospace* (IEEE, 1997), Vol 3, pp. 423–439.
19. A. L. Chan, S. Z. Der, and N. M. Nasarabadi, "Multistage infrared target detection," Opt. Eng. **42**, 2746–2754 (2003).
20. S. Z. Der, A. L. Chan, N. M. Nasarabadi, and H. Kwon, "Automated vehicle detection in forward-looking infrared imagery," Appl. Opt. **43**, 333–348 (2004).
21. A. L. Chan, S. A. Rizvi, and N. M. Nasarabadi, "Dualband FLIR for automatic target recognition," Information Fusion **4**, 35–45 (2003).
22. D. Borghys, P. Verlinde, C. Perneel, and M. Acheroy, "Multilevel data fusion for the detection of targets using multispectral image sequences," Opt. Eng. **37**, 477–484 (1998).
23. W. Lie-Chan, S. Z. Der, and N. M. Nasarabadi, "Automatic target recognition using feature-decomposition and data-decomposition modular neural network," IEEE Trans. Image Process. **7**, 1113–1121 (1998).
24. B. Ernisse, S. K. Rogers, M. P. DeSimio, and R. A. Ranies, "Complete automatic target cuer/recognition system for tactical forward-looking infrared images," Opt. Eng. **36**, 2593–2603 (1997).
25. J. Waldemark, V. Becanovic, T. Lindblad, and C. S. Lindsey, "Hybrid neural networks for automatic target recognition," in *Systems, Man, and Cybernetics, IEEE Int. Conf. Computational Cybernetics and Simulation* (IEEE, 1997), Vol. 4, pp. 4016–4021.
26. S. Rogers, J. Colombi, C. Martin, J. Gainey, K. Fielding, T. Burns, D. Ruck, M. Kabrisky, and M. Oxley, "Neural networks for automatic target recognition," Neural Networks **8**, 1153–1184 (1995).
27. G. A. Carpenter, "Neural network models for pattern recognition and associative memory," Neural Networks **2**, 243–257 (1989).
28. I. E. Dror, M. Zagaeski, and C. F. Moss, "Three-dimensional target recognition via sonar: a neural network model," Neural Networks **8**, 149–160 (1995).
29. A. Ravichandran and B. Yegnanarayana, "Studies on object recognition from degraded images using neural networks," Neural Networks **8**, 481–488 (1995).
30. B. Bhanu and T. Jones, "Image understanding research for automatic target recognition," in *IEEE Trans. Aerospace and Electronic Systems Magazine* (IEEE, 1993), pp. 15–22.
31. S. A. Rizvi and N. M. Nasarabadi, "A modular clutter rejection technique for FLIR imagery using region based principal component analysis," Pattern Recogn. **35**, 2895–2904 (2002).
32. S. A. Rizvi and N. M. Nasarabadi, "Fusion of FLIR automatic target recognition algorithms," Information Fusion **4**, 247–258 (2003).
33. J. F. Khan and M. S. Alam, "Target detection in cluttered forward-looking infrared imagery," Opt. Eng. **44**, 076404 (2005).
34. R. C. Gonzalez and R. E. Woods, *Digital Image Processing* (Addison-Wesley, 1992).
35. T. C. Wang and N. B. Karayiannis, "Detection of microcalcifications in digital mammograms using wavelets," IEEE Trans. Med. Imaging **17** (1998).
36. I. Daubechies, *Ten Lectures on Wavelets* (SIAM, 1992).
37. C. K. Chui, *An Introduction to Wavelets* (Academic, 1992).
38. G. Strang and T. Nguyen, *Wavelets and Filter Banks* (Wellesley Cambridge, 1996).
39. D. Davies, P. Palmer, and M. Mirmehdi, "Detection and tracking of very small low contrast objects," in *Proceedings of the 9th British Machine Vision Conference* (BMVA, 1998), pp. 599–608.
40. I. Daubechies, "Orthonormal bases of compactly supported wavelets," Commun. Pure Appl. Math. **41**, 909–996 (1988).
41. K. Messer, D. de Ridder, and J. Kittler, "Adaptive texture representation methods for automatic target recognition," in *Proceedings of the 10th British Machine Vision Conference* (BMVA, 1999), pp. 443–452.
42. D. F. Specht, "Probabilistic neural networks," Neural Networks **3**, 109–118 (1990).
43. G. S. Gill and J. S. Sohal, "Battlefield decision making: a neural network approach," J. Theoret. Appl. Inform. Technol. **4**, 697–699 (2008).
44. T. Sando, "Modeling highway crashes using Bayesisn belief networks and GIS," Ph.D dissertation (Florida State University, 2005).