Image Recognition and Classification

Algorithms, Systems, and Applications

edited by
Bahram Javidi

Taylor & Francis
Taylor & Francis Group
http://taylorandfrancis.com

Image Recognition and Classification

OPTICAL ENGINEERING

Founding Editor

Brian J. Thompson

Distinguished University Professor
Professor of Optics
Provost Emeritus
University of Rochester
Rochester, New York

Editorial Board

Toshimitsu Asakura
Hokkai-Gakuen University
Sapporo, Hokkaido, Japan

Nicholas F. Borrelli
Corning, Inc.
Corning, New York

Chris Dainty
Imperial College of Science,
Technology, and Medicine
London, England

Bahram Javidi
University of Connecticut
Storrs, Connecticut

Mark Kuzyk
Washington State University
Pullman, Washington

Hiroshi Murata
The Furukawa Electric Co., Ltd.
Yokohama, Japan

Edmond J. Murphy
JDS/Uniphase
Bloomfield, Connecticut

Dennis R. Pape
Photonic Systems Inc.
Melbourne, Florida

Joseph Shamir
Technion–Israel Institute
of Technology
Hafai, Israel

David S. Weiss
Heidelberg Digital L.L.C.
Rochester, New York

1. Electron and Ion Microscopy and Microanalysis: Principles and Applications, *Lawrence E. Murr*
2. Acousto-Optic Signal Processing: Theory and Implementation, *edited by Norman J. Berg and John N. Lee*
3. Electro-Optic and Acousto-Optic Scanning and Deflection, *Milton Gottlieb, Clive L. M. Ireland, and John Martin Ley*
4. Single-Mode Fiber Optics: Principles and Applications, *Luc B. Jeunhomme*
5. Pulse Code Formats for Fiber Optical Data Communication: Basic Principles and Applications, *David J. Morris*
6. Optical Materials: An Introduction to Selection and Application, *Solomon Musikant*
7. Infrared Methods for Gaseous Measurements: Theory and Practice, *edited by Joda Wormhoudt*
8. Laser Beam Scanning: Opto-Mechanical Devices, Systems, and Data Storage Optics, *edited by Gerald F. Marshall*
9. Opto-Mechanical Systems Design, *Paul R. Yoder, Jr.*
10. Optical Fiber Splices and Connectors: Theory and Methods, *Calvin M. Miller with Stephen C. Mettler and Ian A. White*
11. Laser Spectroscopy and Its Applications, *edited by Leon J. Radziemski, Richard W. Solarz, and Jeffrey A. Paisner*
12. Infrared Optoelectronics: Devices and Applications, *William Nunley and J. Scott Bechtel*
13. Integrated Optical Circuits and Components: Design and Applications, *edited by Lynn D. Hutcheson*
14. Handbook of Molecular Lasers, *edited by Peter K. Cheo*
15. Handbook of Optical Fibers and Cables, *Hiroshi Murata*
16. Acousto-Optics, *Adrian Korpel*
17. Procedures in Applied Optics, *John Strong*
18. Handbook of Solid-State Lasers, *edited by Peter K. Cheo*
19. Optical Computing: Digital and Symbolic, *edited by Raymond Arrathoon*
20. Laser Applications in Physical Chemistry, *edited by D. K. Evans*
21. Laser-Induced Plasmas and Applications, *edited by Leon J. Radziemski and David A. Cremers*
22. Infrared Technology Fundamentals, *Irving J. Spiro and Monroe Schlessinger*
23. Single-Mode Fiber Optics: Principles and Applications, Second Edition, Revised and Expanded, *Luc B. Jeunhomme*
24. Image Analysis Applications, *edited by Rangachar Kasturi and Mohan M. Trivedi*
25. Photoconductivity: Art, Science, and Technology, *N. V. Joshi*
26. Principles of Optical Circuit Engineering, *Mark A. Mentzer*
27. Lens Design, *Milton Laikin*
28. Optical Components, Systems, and Measurement Techniques, *Rajpal S. Sirohi and M. P. Kothiyal*
29. Electron and Ion Microscopy and Microanalysis: Principles and Applications, Second Edition, Revised and Expanded, *Lawrence E. Murr*

30. Handbook of Infrared Optical Materials, *edited by Paul Klocek*
31. Optical Scanning, *edited by Gerald F. Marshall*
32. Polymers for Lightwave and Integrated Optics: Technology and Applications, *edited by Lawrence A. Hornak*
33. Electro-Optical Displays, *edited by Mohammad A. Karim*
34. Mathematical Morphology in Image Processing, *edited by Edward R. Dougherty*
35. Opto-Mechanical Systems Design: Second Edition, Revised and Expanded, *Paul R. Yoder, Jr.*
36. Polarized Light: Fundamentals and Applications, *Edward Collett*
37. Rare Earth Doped Fiber Lasers and Amplifiers, *edited by Michel J. F. Digonnet*
38. Speckle Metrology, *edited by Rajpal S. Sirohi*
39. Organic Photoreceptors for Imaging Systems, *Paul M. Borsenberger and David S. Weiss*
40. Photonic Switching and Interconnects, *edited by Abdellatif Marrakchi*
41. Design and Fabrication of Acousto-Optic Devices, *edited by Akis P. Goutzoulis and Dennis R. Pape*
42. Digital Image Processing Methods, *edited by Edward R. Dougherty*
43. Visual Science and Engineering: Models and Applications, *edited by D. H. Kelly*
44. Handbook of Lens Design, *Daniel Malacara and Zacarias Malacara*
45. Photonic Devices and Systems, *edited by Robert G. Hunsperger*
46. Infrared Technology Fundamentals: Second Edition, Revised and Expanded, *edited by Monroe Schlessinger*
47. Spatial Light Modulator Technology: Materials, Devices, and Applications, *edited by Uzi Efron*
48. Lens Design: Second Edition, Revised and Expanded, *Milton Laikin*
49. Thin Films for Optical Systems, *edited by François R. Flory*
50. Tunable Laser Applications, *edited by F. J. Duarte*
51. Acousto-Optic Signal Processing: Theory and Implementation, Second Edition, *edited by Norman J. Berg and John M. Pellegrino*
52. Handbook of Nonlinear Optics, *Richard L. Sutherland*
53. Handbook of Optical Fibers and Cables: Second Edition, *Hiroshi Murata*
54. Optical Storage and Retrieval: Memory, Neural Networks, and Fractals, *edited by Francis T. S. Yu and Suganda Jutamulia*
55. Devices for Optoelectronics, *Wallace B. Leigh*
56. Practical Design and Production of Optical Thin Films, *Ronald R. Willey*
57. Acousto-Optics: Second Edition, *Adrian Korpel*
58. Diffraction Gratings and Applications, *Erwin G. Loewen and Evgeny Popov*
59. Organic Photoreceptors for Xerography, *Paul M. Borsenberger and David S. Weiss*
60. Characterization Techniques and Tabulations for Organic Nonlinear Optical Materials, *edited by Mark Kuzyk and Carl Dirk*

61. Interferogram Analysis for Optical Testing, *Daniel Malacara, Manuel Servín, and Zacarias Malacara*
62. Computational Modeling of Vision: The Role of Combination, *William R. Uttal, Ramakrishna Kakarala, Sriram Dayanand, Thomas Shepherd, Jagadeesh Kalki, Charles F. Lunskis, Jr., and Ning Liu*
63. Microoptics Technology: Fabrication and Applications of Lens Arrays and Devices, *Nicholas F. Borrelli*
64. Visual Information Representation, Communication, and Image Processing, *Chang Wen Chen and Ya-Qin Zhang*
65. Optical Methods of Measurement: Wholefield Techniques, *Rajpal S. Sirohi and Fook Siong Chau*
66. Integrated Optical Circuits and Components: Design and Applications, *edited by Edmond J. Murphy*
67. Adaptive Optics Engineering Handbook, *edited by Robert K. Tyson*
68. Entropy and Information Optics, *Francis T. S. Yu*
69. Computational Methods for Electromagnetic and Optical Systems, *John M. Jarem and Partha P. Banerjee*
70. Laser Beam Shaping: Theory and Techniques, *edited by Fred M. Dickey and Scott C. Holswade*
71. Rare-Earth-Doped Fiber Lasers and Amplifiers: Second Edition, Revised and Expanded, *edited by Michel J. F. Digonnet*
72. Lens Design: Third Edition, Revised and Expanded, *Milton Laikin*
73. Handbook of Optical Engineering, *edited by Daniel Malacara and Brian J. Thompson*
74. Handbook of Imaging Materials, *edited by Arthur S. Diamond and David S. Weiss*
75. Handbook of Image Quality: Characterization and Prediction, *Brian W. Keelan*
76. Fiber Optic Sensors, *edited by Francis T. S. Yu and Shizhuo Yin*
77. Optical Switching/Networking and Computing for Multimedia Systems, *edited by Mohsen Guizani and Abdella Battou*
78. Image Recognition and Classification: Algorithms, Systems, and Applications, *edited by Bahram Javidi*

Additional Volumes in Preparation

Ultrafast Lasers: Technology and Applications, *edited by Martin Fermann, Almantas Galvanauskas, and Gregg Sucha*

Practical Design and Production of Optical Thin Films: Second Edition, Revised and Expanded, *Ronald R. Willey*

Taylor & Francis
Taylor & Francis Group
http://taylorandfrancis.com

Image Recognition and Classification

Algorithms, Systems, and Applications

edited by

Bahram Javidi

University of Connecticut
Storrs, Connecticut

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

First published 2002 by Marcel Dekker, Inc.

Published 2020 by CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2002 by Taylor & Francis Group, LLC

CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works

ISBN 13: 978-0-8247-0783-5 (hbk)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (http://www.copyright.com/) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

DOI: 10.1201/9780203910962

For my Aunt Matin

Taylor & Francis
Taylor & Francis Group
http://taylorandfrancis.com

Preface

Image recognition and classification is one of the most actively pursued areas in the broad field of imaging sciences and engineering. The reason is evident: the ability to replace human visual capabilities with a machine is very important and there are diverse applications. The main idea is to inspect an image scene by processing data obtained from sensors. Such machines can substantially reduce the workload and improve accuracy of making decisions by human operators in diverse fields including the military and defense, biomedical engineering systems, health monitoring, surgery, intelligent transportation systems, manufacturing, robotics, entertainment, and security systems.

Image recognition and classification is a multidisciplinary field. It requires contributions from diverse technologies and expertise in sensors, imaging systems, signal/image processing algorithms, VLSI, hardware and software, and packaging/integration systems.

In the military, substantial efforts and resources have been placed in this area. The main applications are in autonomous or aided target detection and recognition, also known as automatic target recognition (ATR). In addition, a variety of sensors have been developed, including high-speed video, low-light-level TV, forward-looking infrared (FLIR), synthetic aperture radar (SAR), inverse synthetic aperture radar (ISAR), laser radar (LADAR), multispectral and hyperspectral sensors, and three-dimensional sensors. Image recognition and classification is considered an extremely useful and important resource available to military personnel and operations in the areas of surveillance and targeting.

In the past, most image recognition and classification applications have been for military hardware because of high cost and performance demands. With recent advances in optoelectronic devices, sensors, electronic hardware, computers, and software, image recognition and classification systems have become available with many commercial applications.

While there have been significant advances in image recognition and classification technologies, major technical problems and challenges face this field. These include large variations in the inspected object signature due to environmental conditions, geometric variations, aging, and target/sensor behavior (e.g., IR thermal signature fluctuations, reflection angles, etc.). In addition, in many applications the target or object of interest is a small part of a very complex scene under inspection; that is, the distorted target signature is embedded in background noise such as clutter, sensor noise, environmental degradations, occlusion, foliage masking, and camouflage. Sometimes the algorithms are developed with a limited available training data set, which may not accurately represent the actual fluctuations of the objects or the actual scene representation, and other distortions are encountered in realistic applications. Under these adverse conditions, a reliable system must perform recognition and classification in real time and with high detection probability and low false alarm rates. Therefore, progress is needed in the advancement of sensors and algorithms and compact systems that integrate sensors, hardware, and software algorithms to provide new and improved capabilities for high-speed accurate image recognition and classification.

This book presents important recent advances in sensors, image processing algorithms, and systems for image recognition and classification with diverse applications in military, aerospace, security, image tracking, radar, biomedical, and intelligent transportation. The book includes contributions by some of the leading researchers in the field to present an overview of advances in image recognition and classification over the past decade. It provides both theoretical and practical information on advances in the field.

The book illustrates some of the state-of-the-art approaches to the field of image recognition using image processing, nonlinear image filtering, statistical theory, Bayesian detection theory, neural networks, and 3D imaging. Currently, there is no single winning technique that can solve all classes of recognition and classification problems. In most cases, the solutions appear to be application-dependent and may combine a number of these approaches to acquire the desired results.

Image Recognition and Classification provides examples, tests, and experiments on real world applications to clarify theoretical concepts. A bibliography for each topic is also included to aid the reader. It is a practical book, in which the systems and algorithms have commercial applications and can be implemented with commercially available computers, sensors, and processors. The book assumes some elementary background in signal/image processing. It is intended for electrical or computer engineers with interests in signal/image processing, optical engineers, computer scientists, imaging scientists, biomedical engineers, applied physicists, applied mathematicians, defense technologists, and graduate students and researchers in these disciplines.

I would like to thank the contributors, most of whom I have known for many years and are my friends, for their fine contributions and hard work. I also thank Russell Dekker for his encouragement and support, and Eric Stannard for his assistance. I hope that this book will be a useful tool to increase appreciation and understanding of a very important field.

Bahram Javidi

Taylor & Francis
Taylor & Francis Group
http://taylorandfrancis.com

Contents

Part II: Three-Dimensional Image Recognition

Part III: Nonlinear Distortion-Tolerant Image Recognition Systems

Part IV: Commercial Applications of Image Recognition Systems

Contributors

Bir Bhanu Center for Research in Intelligent Systems, University of California, Riverside, California

Francis Chan Naval Undersea Warfare Center, Newport, Rhode Island

Lipchen Alex Chan U.S. Army Research Laboratory, Adelphi, Maryland

Rama Chellappa University of Maryland, College Park, Maryland

Sandor Z. Der U.S. Army Research Laboratory, Adelphi, Maryland

Frank Dubois Université Libre de Bruxelles, Bruxelles, Belgium

Olivier Germain Ecole National Supérieure de Physique de Marseille, Domaine Universitaire de Saint-Jérôme, Marseille, France

Seung Hyun Hong University of Connecticut, Storrs, Connecticut

Bahram Javidi University of Connecticut, Storrs, Connecticut

Grinnell Jones III Center for Research in Intelligent Systems, University of California, Riverside, California

Sherif Kishk University of Connecticut, Storrs, Connecticut

Abhijit Mahalanobis Lockheed Martin, Orlando, Florida

Hesham Mahmoud Wireless Facilities, Inc., San Diego, California

Osamu Matoba Institute of Industrial Science, University of Tokyo, Tokyo, Japan

Christophe Minetti Université Libre de Bruxelles, Bruxelles, Belgium

Nasser M. Nasrabadi U.S. Army Research Laboratory, Adelphi, Maryland

Luting Pan University of Connecticut, Storrs, Connecticut

Elisabet Pérez Polytechnic University of Catalunya, Terrassa, Spain

Brian Redman Arete Associates, Tucson, Arizona

Philippe Réfrégier Ecole National Supérieure de Physique de Marseille, Domaine Universitaire de Saint-Jérôme, Marseille, France

Joseph Rosen Ben-Gurion University of the Negev, Beer-Sheva, Israel

Firooz Sadjadi Lockheed Martin, Saint Anthony, Minnesota

Enrique Tajahuerce Universitat Jaume I, Castellon, Spain

Wen-Yi Zhao Sarnoff Corporation, Princeton, New Jersey

Qinfen Zheng University of Maryland, College Park, Maryland

Image Recognition and Classification

Taylor & Francis
Taylor & Francis Group
http://taylorandfrancis.com

1

Neural-Based Target Detectors for Multiband Infrared Imagery

Lipchen Alex Chan, Sandor Z. Der, and Nasser M. Nasrabadi
U.S. Army Research Laboratory, Adelphi, Maryland

1.1 INTRODUCTION

Human visual performance greatly exceeds computer capabilities, probably because of superior high-level image understanding, contextual knowledge, and massively parallel processing. Human capabilities deteriorate drastically in a low-visibility environment or after an extended period of surveillance, and certain working environments are either inaccessible or too hazardous for human beings. For these reasons, automatic recognition systems are developed for various military and civilian applications. Driven by advances in computing capability and image processing technology, computer mimicry of human vision has recently gained ground in a number of practical applications. Specialized recognition systems are becoming more likely to satisfy stringent constraints in accuracy and speed, as well as the cost of development and maintenance.

The development of robust automatic target recognition (ATR) systems must still overcome a number of well-known challenges: for example, the large number of target classes and aspects, long viewing range, obscured targets, high-clutter background, different geographic and weather conditions, sensor noise, and variations caused by translation, rotation, and scaling of the targets. Inconsistencies in the signature of targets, similarities between the signatures of different targets, limited training and testing data, camouflaged targets, nonrepeatability of target signatures, and

DOI: 10.1201/9780203910962-1

difficulty using available contextual information makes the recognition problem even more challenging.

A complete ATR system typically consists of several algorithmic components, such as preprocessing, detection, segmentation, feature extraction, classification, prioritization, tracking, and aimpoint selection [1]. Among these components, we are particularly interested in the detection-classification modules, which are shown in Fig. 1. To lower the likelihood of omitting targets of interest, a detector must accept a nonzero false-alarm rate. Figure 1 shows the output of a detector on a typical image. The detector has found the target but has also selected a number of background regions as potential targets. To enhance the performance of the system, an explicit clutter rejector may be added to reject most of the false alarms produced by the detector while eliminating only a few of the targets. Clutter rejectors tend to be much more complex than the detector, giving better performance at the cost of greater computational complexity. The computational cost is often unimportant because the clutter rejector needs to operate only on the small subset of the image that is indicated by the detector.

The ATR learning environment, in which the training data are collected, exerts a powerful influence on the design and performance of an ATR system. Dasarathy [2] described these environments in an increasing order of difficulty, namely the supervised, imperfectly supervised, unfamiliar, vicissitudinous, unsupervised, and partially exposed environments. In this chapter, we assume that our training data come from an unfamiliar environment, where the labels of the training data might be unreliable to a level that is not known a priori. For the experimentation presented in this chapter, the input images were obtained by forward-looking infrared

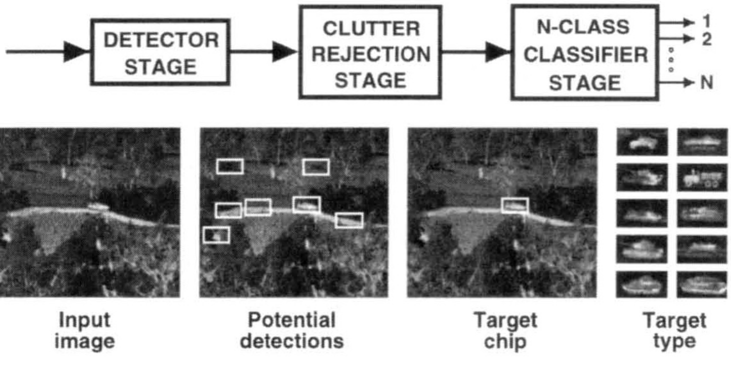

Figure 1 A typical ATR system.

(FLIR) sensors. For these sensors, the signatures of the targets within the scene are severely affected by rain, fog, and foliage [3]. Clark et al. [4] used an information-theoretic approach to evaluate the information bound of FLIR images in order to estimate the best possible performance of any ATR algorithm that uses the given FLIR images as inputs. On the other hand, some FLIR enhancement techniques may be used to preprocess the FLIR input images. Lo [5] examined six of these techniques and found that a variable threshold zonal filtering technique performed most satisfactorily.

The major goal of this research is to examine the benefits of using two passive infrared images, sensitive to different portions of the spectrum, as inputs to a target detector and clutter rejector. The two frequency bands that we use are normally described as mid-wave (MW, 3–5 μm) and long-wave (LW, 8–12 μm) infrared. Two such images are shown in Fig. 2. Although these images look roughly similar, there are places where different intensities can be noted. The difference tends to be more significant during the day, because reflected solar energy is significant in the mid-wave band, but not in the long-wave band. These differences have indeed affected the detection results of an automatic target detector. As shown in Fig. 3, different regions of interest were identified by the same target detector on these two images. Because a different performance is obtained using either the MW or the LW imagery, our first question is which band alone provides better performance in target detection and clutter rejection? The second question is whether combining the bands results in better performance than using either band alone, and if so, what are the best methods of combining these two bands.

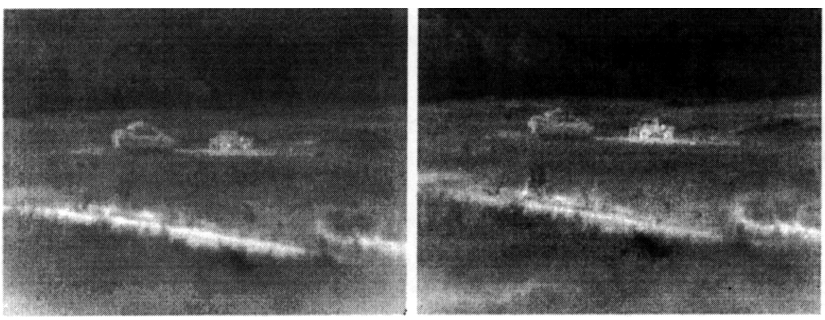

Figure 2 Typical FLIR images for the mid-wave (left) and long-wave (right) bands, with an M2 tank and a HMMWV around the image center. Different degree of radiation, as shown by the windshield of the HMMWV, is quite apparent.

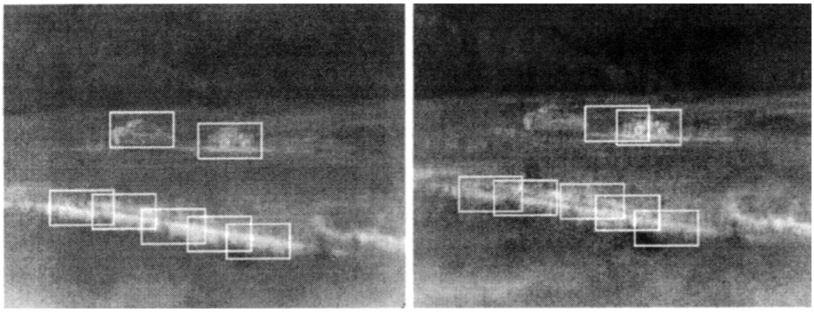

Figure 3 The first seven regions of interest detected on the mid-wave (left) and the long-wave (right) bands. Note that the M2 tank is missed in the case of the long-wave image but detected in the mid-wave image.

To answers these questions, we developed a set of eigen-neural-based modules and use them as either a target detector or clutter rejector in our experiments. As shown in Fig. 4, our typical detector/rejector module consists of an eigenspace transformation and a multilayer perceptron (MLP). The input to the module is the region of interest (target chip) extracted either from an individual band or from both of the MW and LW bands simultaneously. An eigen transformation is used for feature extraction and dimensionality reduction. The transformations considered in this chapter are principal component analysis (PCA) [6], the eigenspace separation transform (EST) [7], and their variants that were jointly optimized with the MLP. These transformations differ in their capability to enhance class separability and to extract component features from a training set. When both bands are input together, the two input chips are transformed through either a set of jointly obtained eigenvectors or two sets of band-specific eigenvectors. The result of the eigenspace transformation is then fed to the MLP that predicts the identity of the input, which is either a target or clutter. Further descriptions about the eigenspace transformation and the MLP are provided in the next two sections. Experimental results are presented in Section 4. Some conclusions are given in the final section of this chapter.

1.2 EIGENTARGETS

We used two methods to obtain the eigentargets from a given set of training chips. PCA is the most basic method, from which the more complicated EST method is derived.

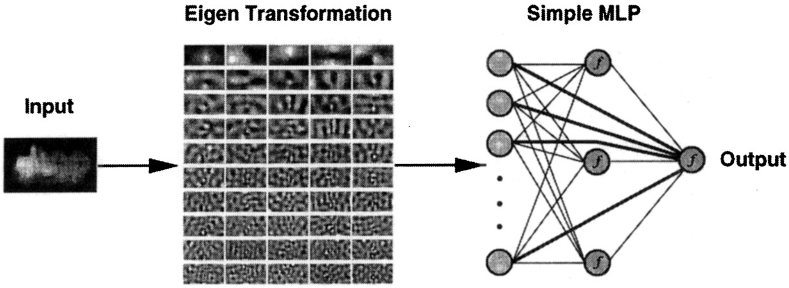

Figure 4 Schematic diagram of our detector/rejector module.

1.2.1 Principal Component Analysis

Also referred to as the Hotelling transform or the discrete Karhunen–Loève transform, PCA is based on statistical properties of vector representations. PCA is an important tool for image processing because it has several useful properties, such as decorrelation of data and compaction of information (energy) [8]. Here, we provide a summary of the basic theory of PCA.

Assume a population of random vectors of the form

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{1}$$

The *mean vector* and the *covariance matrix* of the vector population **x** are defined as

$$\mathbf{m_x} = E\{\mathbf{x}\} \tag{2}$$

$$\mathbf{C_x} = E\{(\mathbf{x} - \mathbf{m_x})(\mathbf{x} - \mathbf{m_x})^T\} \tag{3}$$

respectively, where $E\{\text{arg}\}$ is the expected value of the argument and T indicates vector transposition. Because **x** is n dimensional $\mathbf{C_x}$ is a matrix of order $n \times n$. Element c_{ii} of $\mathbf{C_x}$ is the variance of x_i (the ith component of the **x** vectors in the population) and element c_{ij} of $\mathbf{C_x}$ is the covariance between elements x_i and x_j of these vectors. The matrix $\mathbf{C_x}$ is real and symmetric. If elements x_i and x_j are uncorrelated, their covariance is zero and, therefore, $c_{ij} = c_{ji} = 0$. For N vector samples from a random population, the mean vector and covariance matrix can be approximated respectively from the samples by

$$\mathbf{m}_{\mathbf{x}} = \frac{1}{N}\sum_{p=1}^{N}\mathbf{x}_p \tag{4}$$

$$\mathbf{C}_{\mathbf{x}} = \frac{1}{N}\sum_{p=1}^{N}(\mathbf{x}_p\mathbf{x}_p^T - \mathbf{m}_{\mathbf{x}}\mathbf{m}_{\mathbf{x}}^T) \tag{5}$$

Because $\mathbf{C}_{\mathbf{x}}$ is real and symmetric, we can always find a set of n orthonormal eigenvectors for this covariance matrix. A simple but sound algorithm to find these orthonormal eigenvectors for all really symmetric matrices is the Jacobi method [9]. The Jacobi algorithm consists of a sequence of orthogonal similarity transformations. Each transformation is just a plane rotation designed to annihilate one of the off-diagonal matrix elements. Successive transformations undo previously set zeros, but the off-diagonal elements get smaller and smaller, until the matrix is effectively diagonal (to the precision of the computer). The eigenvectors are obtained by accumulating the product of transformations during the process, and the main diagonal elements of the final diagonal matrix are the eigenvalues. Alternatively, a more complicated method based on the QR algorithm for real Hessenberg matrices can be used [9]. This is a more general method because it can extract eigenvectors from a nonsymmetric real matrix. It becomes increasingly more efficient than the Jacobi method as the size of the matrix increases. Because we are dealing with large matrices, we used the QR method for all experiments described in this chapter. Figure 5 shows the first 100 (out of the 800 possible in this case) most dominant PCA eigentargets and eigenclutters, which were extracted from the target and clutter chips in the training set, respectively. Having the largest eigenvalues, these eigenvectors capture the greatest variance or energy as well as the most meaningful features among the training data.

Let $\mathbf{e}_i$ and λ_i, $i = 1, 2, \ldots, n$, be the eigenvectors and the corresponding eigenvalues, respectively, of $\mathbf{C}_{\mathbf{x}}$, sorted in a descending order so that $\lambda_j \geq \lambda_{j+1}$ for $j = 1, 2, \ldots, n-1$. Let $\mathbf{A}$ be a matrix whose rows are formed from the eigenvectors of $\mathbf{C}_{\mathbf{x}}$, such that

$$\mathbf{A} = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \vdots \\ \mathbf{e}_n \end{bmatrix} \tag{6}$$

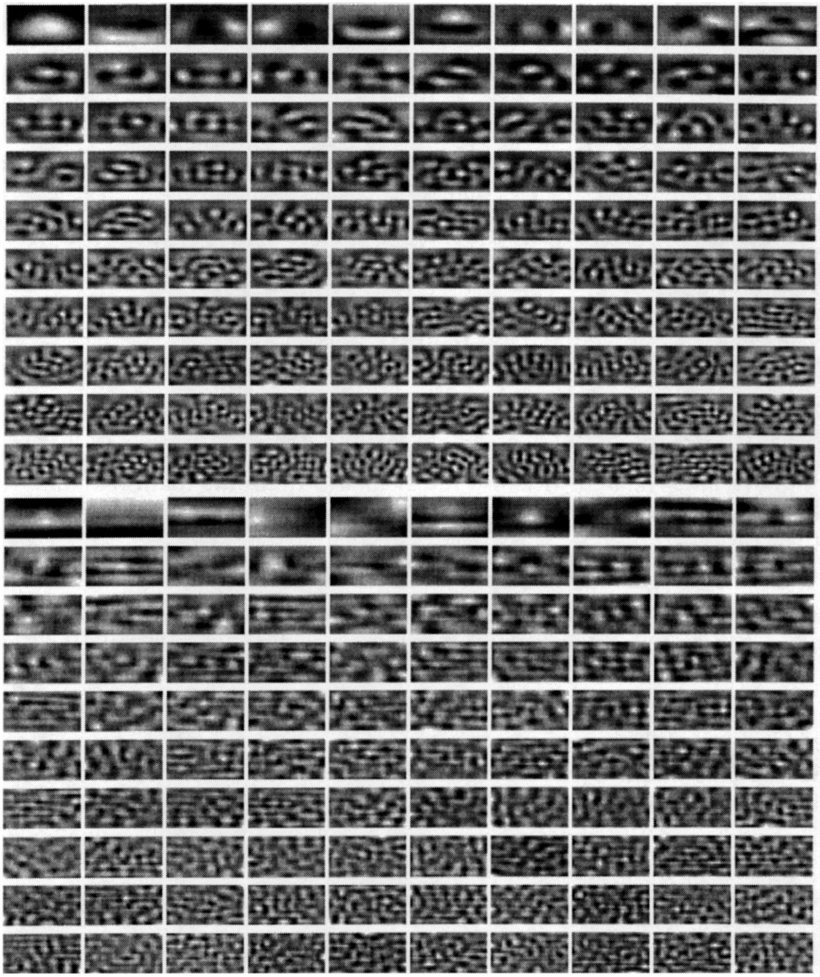

Figure 5 First 100 most dominant PCA eigenvectors extracted from the target (top) and clutter (bottom) chips.

This $\mathbf{A}$ matrix can be used as a linear transformation matrix that maps the x's into vectors, denoted by $\mathbf{y}$'s, as follows:

$$\mathbf{y} = \mathbf{A}(\mathbf{x} - \mathbf{m_x}) \tag{7}$$

The y vectors resulting from this transformation have a zero mean vector; that is, $\mathbf{m_y} = 0$. The covariance matrix of the $\mathbf{y}$'s can be computed from $\mathbf{A}$ and $\mathbf{C_x}$ by

$$\mathbf{C_y} = \mathbf{A}\mathbf{C_x}\mathbf{A}^T \tag{8}$$

Furthermore, $\mathbf{C_y}$ is a diagonal matrix whose elements along the main diagonal are the eigenvalues of $\mathbf{C_x}$; that is,

$$\mathbf{C_y} = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \cdot & \\ & & & \cdot \\ 0 & & & \lambda_n \end{bmatrix} \tag{9}$$

Because the off-diagonal elements of $\mathbf{C_y}$ are zero, the elements of the y vectors are uncorrelated. Because the elements along the main diagonal of a diagonal matrix are its eigenvalues, $\mathbf{C_x}$ and $\mathbf{C_y}$ have the same eigenvalues and eigenvectors.

On the other hand, we may want to reconstruct vector $\mathbf{x}$ from vector y. Because the rows of $\mathbf{A}$ are orthonormal vectors, $\mathbf{A}^{-1} = \mathbf{A}^T$. Therefore, any vector $\mathbf{x}$ can be reconstructed from its corresponding $\mathbf{y}$ by the relation

$$\mathbf{x} = \mathbf{A}^T\mathbf{y} + \mathbf{m_x} \tag{10}$$

Instead of using all the eigenvectors of $\mathbf{C_x}$, we may pick only k eigenvectors corresponding to the k largest eigenvalues and form a new transformation matrix $\mathbf{A}_k$ of order $k \times n$. In this case, the resulting $\mathbf{y}$ vectors would be k dimensional, and the reconstruction given in Eq. (10) would no longer be exact. The reconstructed vector using $\mathbf{A}_k$ is

$$\hat{\mathbf{x}} = \mathbf{A}_k^T\mathbf{y} + \mathbf{m_x} \tag{11}$$

The mean square error (MSE) between $\mathbf{x}$ and $\hat{\mathbf{x}}$ can be computed by the expression

$$\epsilon = \sum_{j=1}^{n}\lambda_j - \sum_{j=1}^{k}\lambda_j = \sum_{j=k+1}^{n}\lambda_j \tag{12}$$

Because the λ_j's decrease monotonically, Eq. (12) shows that we can minimize the error by selecting the k eigenvectors associated with the k largest eigenvalues. Thus, the PCA transform is optimal in the sense that it minimizes the MSE between the vectors $\mathbf{x}$ and their approximations $\hat{\mathbf{x}}$.

1.2.2 Eigenspace Separation Transform

The EST has been proposed by Torrieri as a preprocessor to a neural binary classifier [10]. The goal of the EST is to transform the input patterns into a set of projection values such that the size of a neural classifier is reduced and its generalization capability is increased. The size of the neural network is reduced, because the EST projects an input pattern into an orthogonal subspace of smaller dimensionality. The EST also tends to produce projections with different average lengths for different classes of input and, hence, improves the discriminability between the targets. In short, the EST preserves and enhances the classification information needed by the subsequent classifier. It has been used in a mine-detection task with some success [11].

The transformation matrix S of the EST can be obtained as follows:

1. Computer the $n \times n$ correlation difference matrix

$$\hat{\mathbf{M}} = \frac{1}{N_1}\sum_{p=1}^{N_1} \mathbf{x}_{1p}\mathbf{x}_{1p}^T - \frac{1}{N_2}\sum_{q=1}^{N_2} \mathbf{x}_{2q}\mathbf{x}_{2q}^T \tag{13}$$

where N_1 and $\mathbf{x}_{1p}$ are the number of patterns and the pth training pattern of Class 1, respectively. N_2 and $\mathbf{x}_{2q}$ are similarly related to Class 2 (which is the complement of Class 1).

2. Calculate the eigenvalues of $\hat{\mathbf{M}}$ $\{\lambda_1 | i = 1, 2, \ldots, n\}$.
3. Calculate the sum of the positive eigenvalues

$$E_+ = \sum_{i=1}^{n} \lambda_i \quad \text{if } \lambda_i > 0 \tag{14}$$

and the sum of the absolute values of the negative eigenvalues

$$E_- = \sum_{i=1}^{n} |\lambda_i| \quad \text{if } \lambda_i < 0 \tag{15}$$

(a) If $E_+ > E_-$, then take all the k eigenvectors of $\hat{\mathbf{M}}$ that have positive eigenvalues and form the $n \times k$ matrix $\mathbf{S}$.
(b) If $E_+ < E_-$, then take all the k eigenvectors of $\hat{\mathbf{M}}$ that have negative eigenvalues and form the $n \times k$ matrix $\mathbf{S}$.
(c) If $E_+ = E_-$, then use either subset of eigenvectors to form the matrix $\mathbf{S}$, preferably the smaller subset.

Given the $\mathbf{S}$ transformation matrix, the projection $\mathbf{y}_p$ of an input pattern $\mathbf{x}_p$ is computed as $\mathbf{y}_p = \mathbf{S}^T\mathbf{x}_p$. The $\mathbf{y}_p$, with a smaller dimension (because $k \leq n$) and presumably larger separability between the classes, can then be sent to a neural classifier. Figure 6 shows the eigenvectors associated with the positive and negative eigenvalues of the $\hat{\mathbf{M}}$ matrix that was computed with the target chips as Class 1 and the clutter chips as Class

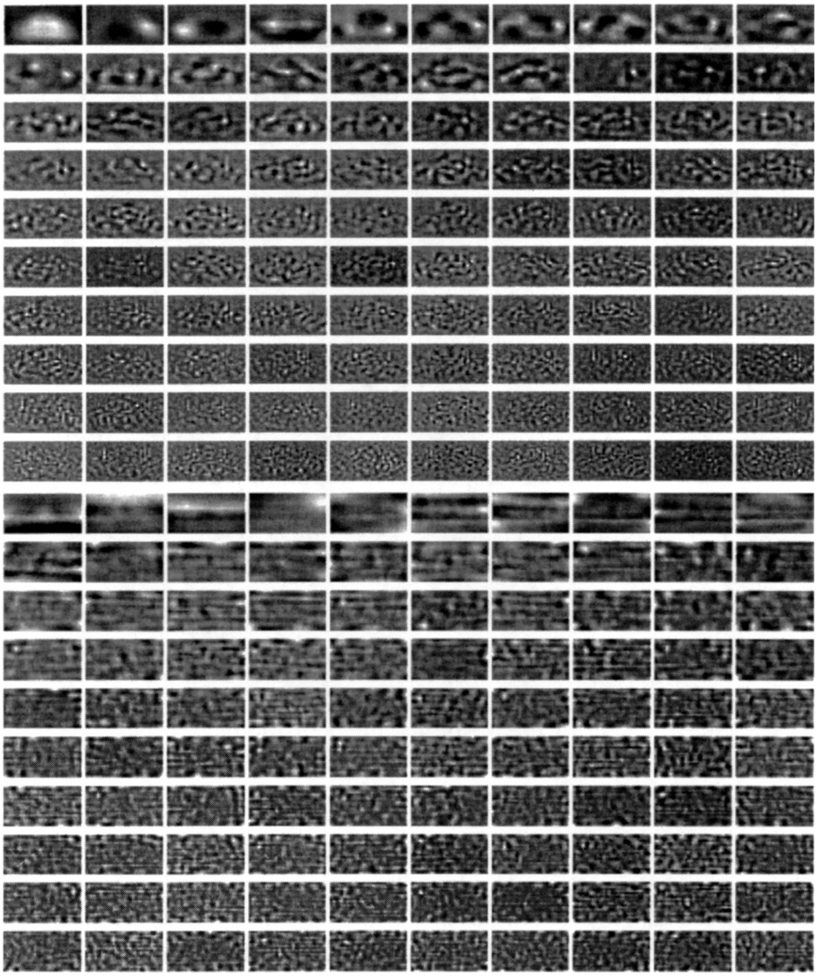

Figure 6 First 100 most dominant EST eigenvectors associated with positive (top) and negative (bottom) eigenvalues.

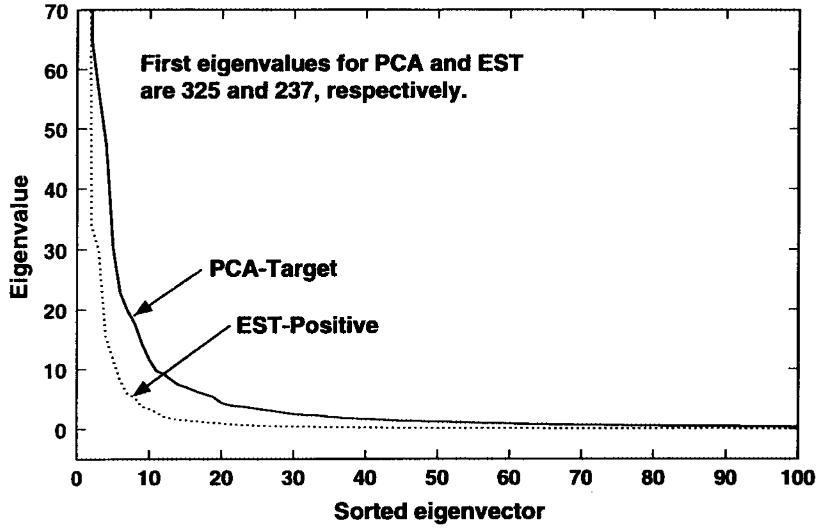

Figure 7 Rapid attenuation of eigenvalues.

2. From the upper part of Fig. 6, the signature of targets can be clearly seen. On the other hand, the lower part represents all the features of clutters.

As we can see from Figs. 5 and 6, only the first few dozens of the eigentargets contain consistent and structurally significant information pertaining to the training data. These eigentargets exhibit a reduction in information content as their associated eigenvalues rapidly decrease, which is depicted in Fig. 7. For the less meaningful eigentargets, say the 50th and all the way up to the 800th, only high-frequency information is present. In other words, by choosing $k = 50$ in Eq. (12) when $n = 800$, the resulting distortion error, ϵ, would be small. Although the distortion is negligible, there is a 16-fold reduction in input dimensionality.

1.3 MULTILAYER PERCEPTRON

After projecting an input chip to a chosen set of k eigentargets, the resulting k projection values are fed to an MLP classifier, where they are combined nonlinearly. A typical MLP used in our experiments, as shown on the right-hand side in Fig. 4, has $k + 1$ input nodes (with an extra bias input), several layers of hidden nodes, and one output node. In addition to full connections between consecutive layers, there are also shortcut connections directly from one layer to all other layers, which may speed up the learning process. The MLP classifier is trained to perform a two-class problem, with training output values of ± 1. Its sole task is to decide whether a given input pattern is a target (indicated by a high output value of around $+1$) or clutter (indicated by a low output value of around -1). The MLP is trained in batch mode using Qprop [12], a modified backpropagation algorithm, for a faster but stable learning course.

Alternatively, the eigenspace transformation can be implemented as an additional linear layer that attaches to the input layer of the simple MLP above. As shown in Fig. 8, the resulting augmented MLP classifier, which is collectively referred to as a PCAMLP network in this chapter, consists of a transformation layer and a back-end MLP (BMLP). When the weights connecting the new input nodes to the kth output node of the transformation layer are initialized with the kth PCA or EST eigenvector, the linear summation at the kth transformation output node is equivalent to the kth projection value. The advantage of this augmented structure is to enable a joint optimization between the transformation (feature extraction) layer and the BMLP classifier, which is achieved by adjusting the corresponding weights of the transformation layer based on the error signals backpropagated from the BMLP classifier.

The purpose of joint optimization is to incorporate class information in the design of the transformation layer. This enhancement is especially

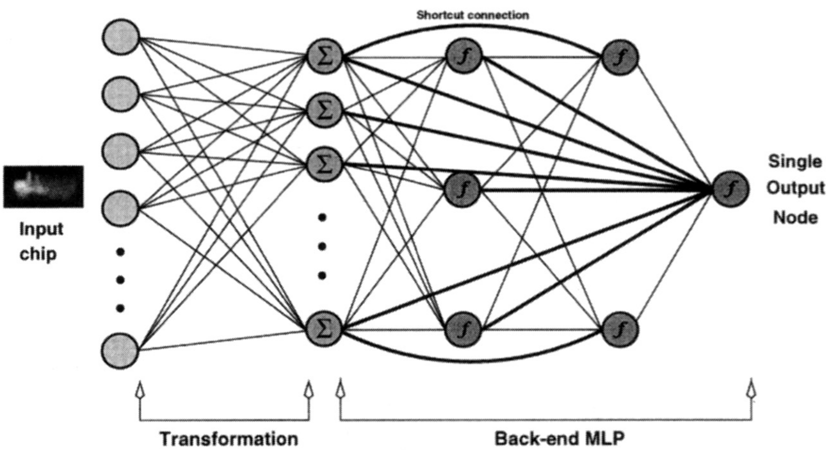

Figure 8 An augmented MLP (or PCAMLP) that consists of a transformation layer and a back-end MLP.

useful to the PCA eigenvectors, because the class-separation issue has never been considered during their derivation. During the joint operation process, the transformation weights are gradually adjusted, suing a variety of gradient descent-based algorithms, so that the overall error is reduced at the output node of the back-end MLP. Although the discriminability of the transformation layer is enhanced, it may lose some of its energy compaction capability in exchange. These changes are exhibited in Fig. 9, where the structural characteristics of the PCA eigenvectors are gradually given away to local emphases that distinguish the targets from clutter. After a prolonged joint optimization process, the succinct PCA structures could be completely replaced by incomprehensible patterns that have overfitted the training samples. Care should be taken to avoid overtraining the transformation layer.

It is interesting to observe that similar evolutions also occur when we initialize the transformation layer with random weights, instead of initializing with the PCA or EST eigenvectors. Adjusted through a supervised gradient descent algorithm, these random weights connected to each output node of the transformation layer gradually evolve into certain features that try to maximize the class separation for the BMLP classifier. A typical evolution of a five-node supervised transformation matrix is shown in Fig. 10, after it had been trained for 689, 3690, 4994, and 9987 epochs, respectively. Note that the random weights at the early stage evolved into more structural features that resemble those of the PCA eigenvectors shown

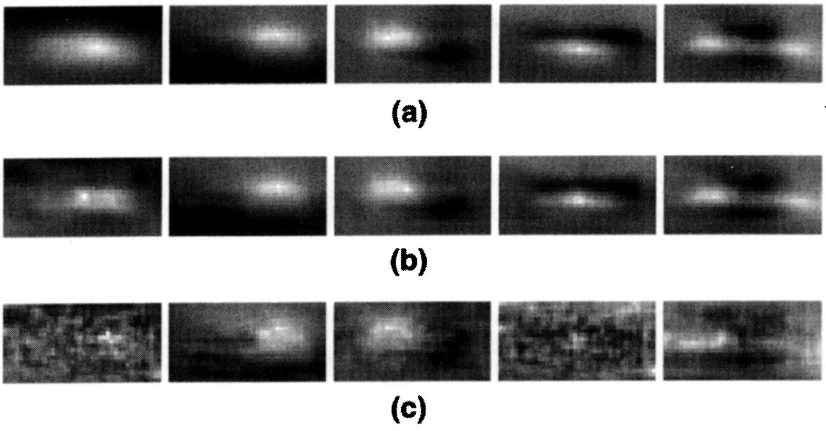

Figure 9 Changes in PCA eigenvectors after (a) 0, (b) 4752, and (c) 15751 epochs of backpropagation training to enhance their discriminability.

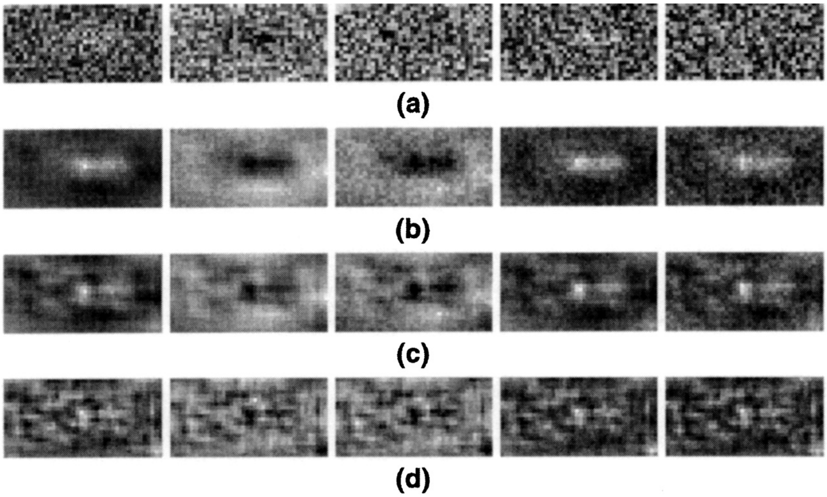

Figure 10 The evolution of transformation vectors that were initialized with random weights and trained with a gradient descent algorithm, after (a) 689, (b) 3690, (c) 4994, and (d) 9987 epochs of training.

in Fig. 9a. Nonetheless, these features became incomprehensible and less structural again when the training session was extended.

In contrast to the PCA transformation, the above supervised transformation does not attempt to optimize the energy compaction on the training data. In addition, the gradient descent algorithm is very likely to be trapped at a local minimum in the treacherous weight space of $p \times m$ dimensions or in its attempts to overfit the training data with strange and spurious solutions. A better approach would be using a more sophisticated training algorithm that is capable of optimizing both the interclass discriminability and energy compaction simultaneously.

Let us first consider the issue of energy compaction during joint discrimination-compression optimization training. Instead of extracting the PCA eigenvectors from the covariance matrix $\mathbf{C_x}$, we can compute them directly from the $\mathbf{x}$ input vectors via a single-layer self-organized neural network [13]. An example of such a neural network, with predefined p input nodes and m linear output nodes, is shown in Fig. 11. If the network is trained with the generalized Hebbian algorithm (GHA) proposed by Sanger [14], the activation value of the kth output neuron, y_k, converges to the kth most dominant eigenvalue associated with the input data. At the same time, the p weights leading to the kth output neuron, w_{ki}, $i = 1, \ldots, p$,

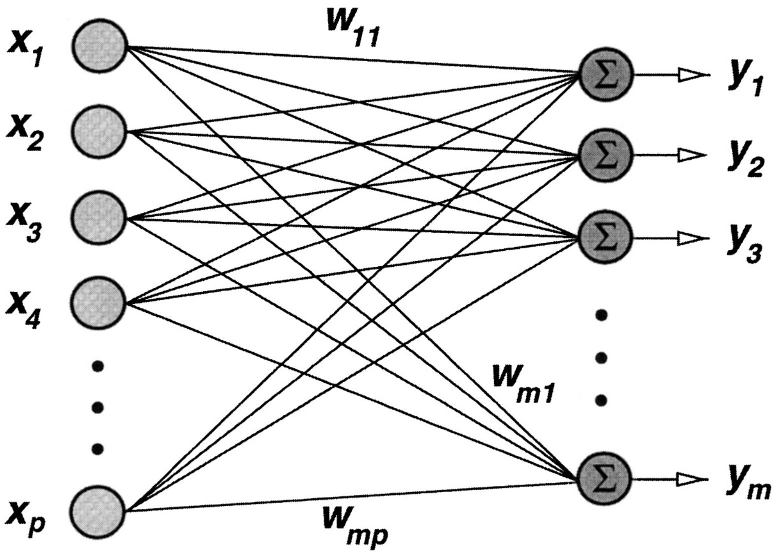

Figure 11 A single-layer self-organized neural network.

become the eigenvector associated with the kth dominant eigenvalue. Suppose we want to find the m most dominant eigenvalues and their associated eigenvectors based on S input samples of size p, namely x_i^s, $s = 1, \ldots, S$, $i = 1, \ldots, p$. The corresponding GHA network can be trained through the following steps:

1. At iteration $t = 1$, initialize all the adjustable weights, w_{ji}, $j = 1, \ldots, m$, $i = 1, \ldots, p$, to small random values. Choose a small positive value for the learning rate parameter η.
2. Compute the output value $y_j^s(t)$ and weight adjustment $\Delta w_{ji}^s(t)$ for $s = 1, \ldots, S, j = 1, \ldots, m$, $i = 1, \ldots, p$, as follows:

$$y_j^s(t) = \sum_{i=1}^{p} w_{ji}(t) x_i^s \tag{16}$$

$$\Delta w_{ji}^s(t) = \eta y_j^s(t) \left(x_i^s - \sum_{k=1}^{j} w_{ki}(t) y_k^s(t) \right) \tag{17}$$

3. Modify the weights, w_{ji}, $j = 1, \ldots, m$, $i = 1, \ldots, p$ for this iteration:

$$w_{ji}(t+1) = w_{ji}(t) + \frac{1}{S}\sum_{s=1}^{S} \Delta w_{ji}^{s}(t) \tag{18}$$

4. Increment t by 1 and go back to Step 2. Repeat Steps 2–4 until all the weights reach their steady-state values.

We combine the unsupervised GHA with a supervised gradient descent algorithm (such as the Qprop algorithm) to perform a joint discrimination–compression optimization. Note that the GHA network in Fig. 11 structurally and functionally resembles the transformation layer of the PCAMLP shown in Fig. 8. Therefore, we may adjust the weights of the transformation layer in Fig. 8 as follows:

$$\begin{aligned} w_{ji}(t+1) &= w_{ji}(t) + \alpha\ [\text{PCA contribution}] + \beta\ [\text{BMLP contribution}] \\ &= w_{ji}(t) + \alpha\left(\frac{1}{S}\sum_{s=1}^{S} \Delta w_{ji}^{s}(t)\right) - \beta\left(\frac{1}{S}\sum_{s=1}^{S} x_i^s \delta_j^s(t)\right) \qquad (19) \\ &= w_{ji}(t) + \frac{1}{S}\sum_{s=1}^{S} \left[\alpha \Delta w_{ji}^{s}(t) - \beta x_i^s \delta_j^s(t)\right] \qquad (20) \end{aligned}$$

The PCA contribution in Eq. (19) is defined earlier as the second term on the right-hand side of Eq. (18). The $\delta_j^s(t)$ in Eq. (20) is the error signal backpropagated from the BMLP to the jth output neuron of the transformation layer for training sample s at iteration t, whereas the x_i^s is the same input vector defined in Eq. (16). The strength of the PCA contribution on the joint transformation is controlled by α, whereas β controls the contribution of gradient descent learning. If $\alpha = 0$, a regular supervised transformation is performed. Setting $\beta = 0$ results in a standard PCA transformation, provided that the η in Eq. (17) is small enough [14].

For the joint transformation to acquire PCA-like characteristics, the η in Eq. (17) and α in Eq. (20) must be small. To prevent the gradient descent effect from dominating the joint transformation, the β has to be small also. As a result, the training process is slow. To speed up the process, we first obtain the standard PCA eigenvectors using the much more efficient QR algorithm [9] and initialize the transformation layer in Fig. 8 with these eigenvectors. Equation (20) is then used to jointly optimize the transformation layer and the classifier together. It is easier to observe performance changes in this way, as the joint transformation attempts to maximize its discriminative power while maintaining its energy compression capability simultaneously.

The effect of this joint discrimination–compression optimization can be clearly seen in Fig. 12. Figure 12a shows the first five most dominant

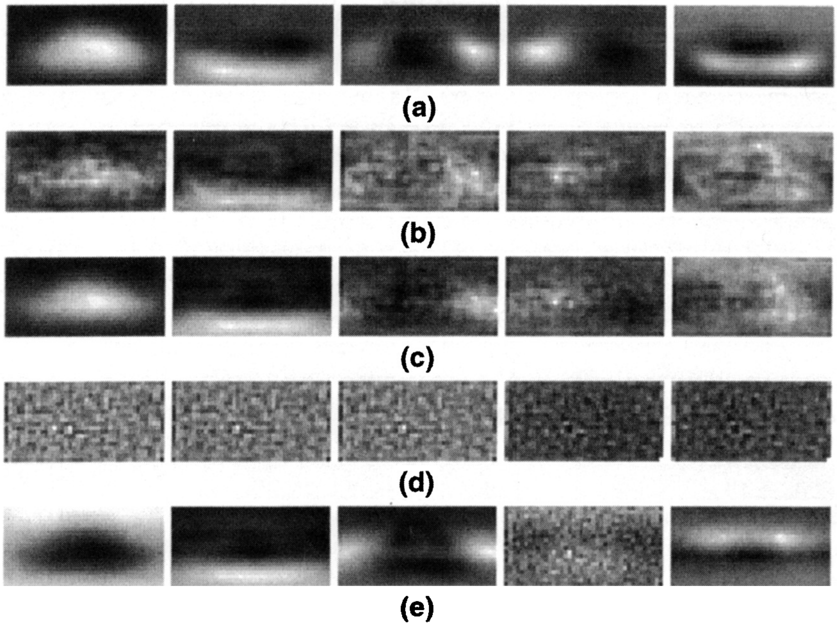

Figure 12 The effect of joint discrimination–compression optimization. The five transformation vectors show as standard PCA eigenvectors (a), after 12519 epochs of Qprop (b), or after 12217 epochs of Qprop + GHA training (c). With randomly initialized values, they appear after 17654 epochs of Qprop (d) or 34788 epochs of Qprop + GHA training (e).

PCA eigenvectors obtained with the standard QR algorithm. If we initialize the transformation layer of the PCAMLP with these standard PCA eigenvectors and adjust them based on the supervised Qprop algorithm only, the resulting weight vectors, as shown in Fig. 12b and similarly in Fig. 9c, would gradually lose all of their succinct structures to quasirandom patterns. However, if Eq. (20) with small nonzero α and β are used, the most important structures of the PCA eigenvectors are always preserved, as we can see in Fig. 12c. If we initialize the transformation vectors with random weights rather than PCA eigenvectors, the Qprop algorithm alone could only forge them into incomprehensible features, as shown in Fig. 12d as well as Fig. 10d, after an extended period of training. With the joint discrimination–compression optimization, even the random weights evolve into the mostly understandable features as shown in Fig. 12e. Out of the five feature vectors displayed in Fig. 12e, only the fourth one fails to exhibit a clear structure. Comparing the other four vectors of Fig. 12e to the corresponding vectors in Fig. 12a, a clear relationship can be established. Reverse-video of the first vector and fifth vector might be caused by an α value that is too large or might be an anomaly of the GHA algorithm when initialized with random weights. The sign of both $w_{ki}(t)$ and $y_k^s(t)$ can flip without affecting the convergence of the algorithm, as can be seen in Eq. (17). The only effect on the back end of the MLP is to flip the signs of the weights that are connected to the $y_k^s(t)$. The other minor differences in these vector pairs are probably the work of the Qprop algorithm.

1.4 EXPERIMENTAL RESULTS

A series of experiments was used to examine the performance of the PCAMLP, either as a target detector or clutter rejector. We also investigated the usefulness of a dual-band FLIR input dataset and the best way to combine the two bands in order to improve the PCAMLP target detector or clutter rejector. We used 12-bit gray-scale FLIR input frames similar to those shown in Fig. 2, each of which measured 500×300 pixels in size. There were 461 pairs of LW–MW matching frames, with 572 legitimate targets posed between 1 and 4 km in each band. First, we trained and tested the PCAMLP as a clutter rejector that processed the output of an automatic target detector called NVDET (developed at the U.S. Army Research Laboratory). Then, we used the trained PCAMLP as a target detector on its own and compared its detection performance to that of NVDET on the same dataset.

1.4.1 PCAMLP as a Clutter Rejector

In order to find the answers for the three questions raised in Section 1.1, we have designed four different clutter rejection setups. As shown in Fig. 13, the first two setups use an individual MW or LW band alone as input. Based on the results from these two setups, we should be able to answer the first question, namely which band alone may perform better in our clutter rejection task? For setup c, we stack the MW and LW chips extracted at the same location before the eigenspace transformations. In this case, the size of each eigenvector is doubled, but not the number of projection values fed to the MLP. If the performance of setup c is better than both setups a and b, then we may say that there is an advantage to using dual band simultaneously. Finally, setup d is almost the same as combining setups a and b, except the

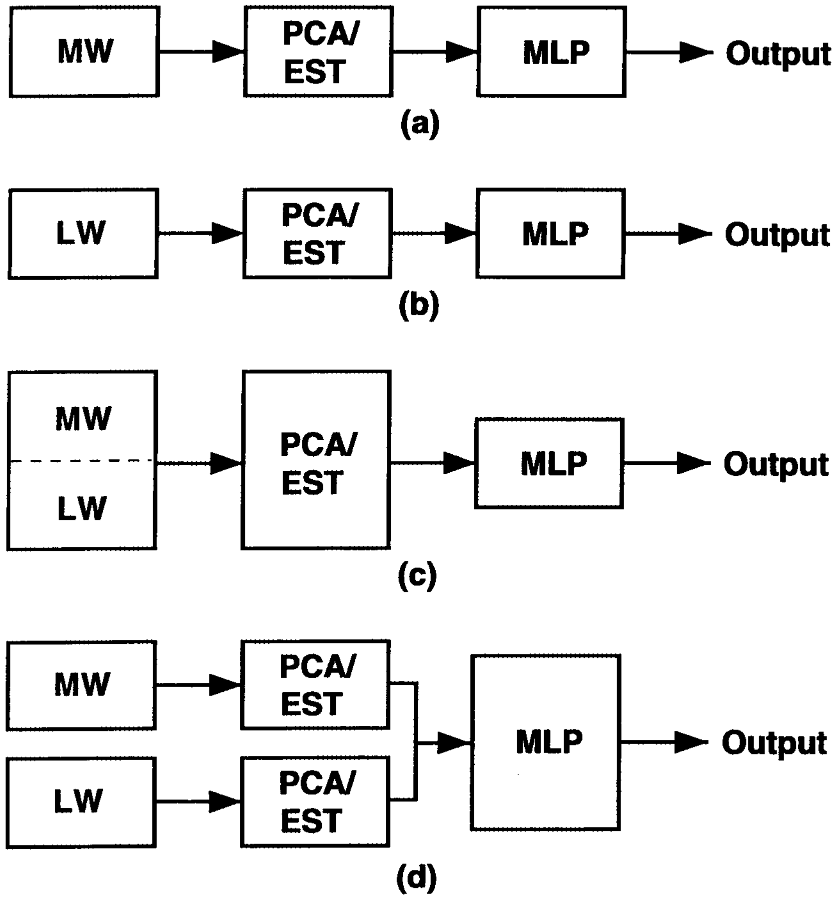

Figure 13 Four different setups for our clutter rejection experiments.

projection values resulting from each eigenspace transformation are now combined before feeding to an MLP with twice as many input nodes. Comparing the performance of setups c and d, we can find out if it is better to combine the two bands before or after the eigenspace transformation.

The chips extracted from each band has a fixed size of 75×40 pixels. Because the range to the targets varies from 1 to 4 km, the size of the targets varies considerably. For the first dataset, the chips were extracted from the location suggested by the NVDET. As shown in Fig. 14, many of these so-

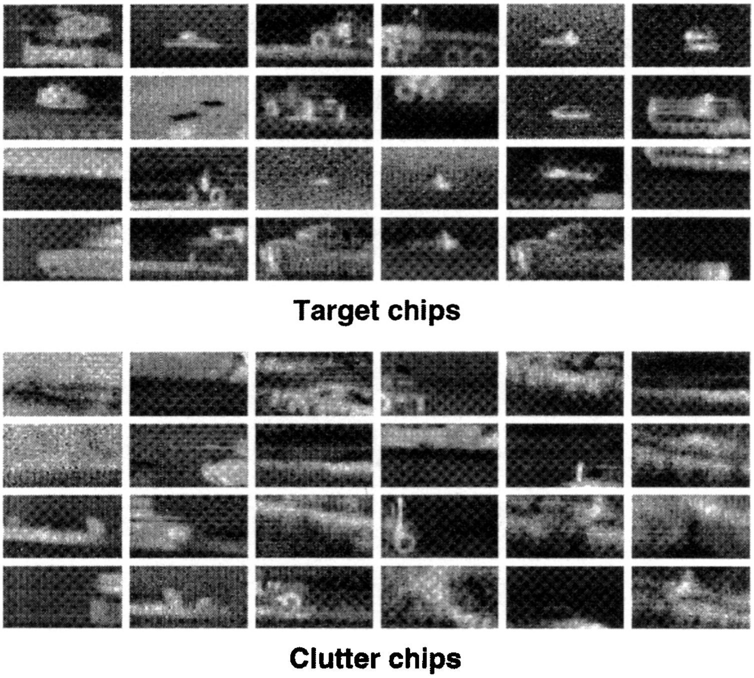

Figure 14 Examples of detector-centered chips.

called detector-centered chips end up with the target lying off-center within the chip. This is a very challenging problem, because the chips of a particular target, posed at the same viewing distance and aspect, may appear different. Furthermore, any detection point would be declared as a miss when its distance from the ground-truth location of a target is greater than a predefined threshold. Hence, a clutter chip extracted around a miss point may contain a significant part of a target, which is very similar to an off-centered target chip. Therefore, it is difficult to find an unequivocal class boundary between the targets and the clutter. The same numbers of chips were created for the MW and LW in all experiments.

We have also created ground-truth-centered chips, which were extracted around the ground truth location of a detected target, as our second dataset. The extraction process of this dataset is almost the same as in the previous dataset, except that whenever a detection suggested by the target detector is declared as an acceptable hit, we move the center of chip extraction from the detected location to the ground-truth center of the corresponding target. In this case, all the target silhouettes were properly centered within the chips, so that a class boundary between the targets and the clutter becomes more feasible. However, some partial targets still appeared on some of the clutter chips, undermining the notion of clear-cut class boundaries. Also, the size of targets continue to fluctuate considerably at different viewing ranges, which complicates the culmination of target distinction. Examples of good-truth-centered chips are given in Fig. 15.

The third dataset consists of chips that were properly centered and zoomed based on ground-truth location and range. The target appears at the center of each chip with a relatively consistent silhouette size. Nonetheless, the signatures of the same target may still exhibit a wide scope of appearances due to differences in zoomed resolution, viewing aspect, operational and weather conditions, environmental effects, and many other factors. Figure 16 shows a few chips from the third dataset.

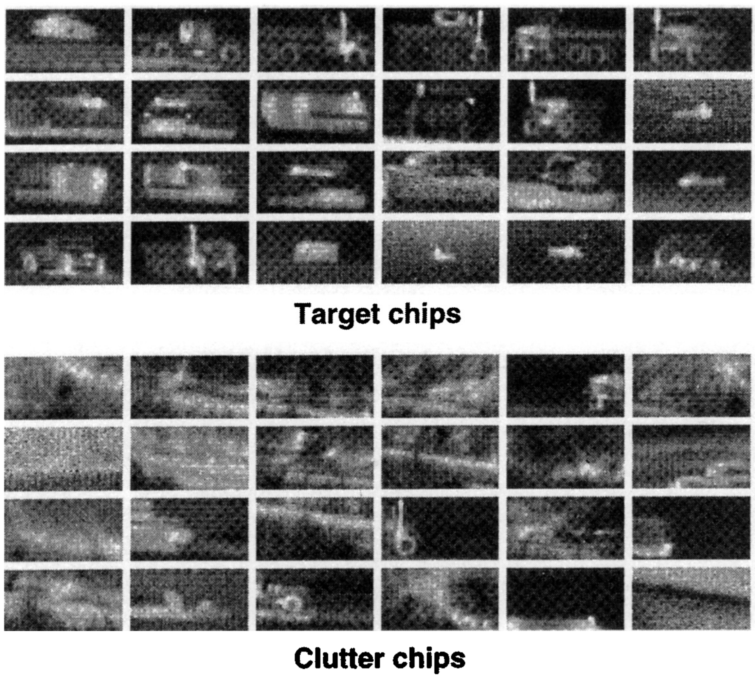

Figure 15 Examples of ground-truth-centered chips.

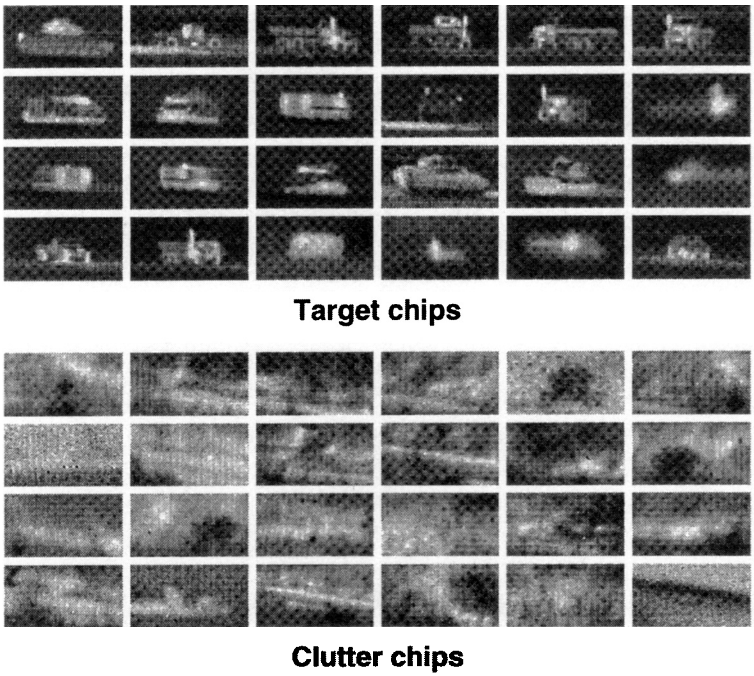

Figure 16 Examples of ground-truth-centered and zoomed chips.

To reduce the computational complexity while retaining enough information embedded in the chips, we down-sampled the input image chip from 75×40 pixels to 40×20 pixels. As shown in Fig. 7, the eigenvalues diminish rapidly for both the PCA and EST methods, but those of the EST decrease even faster. In other words, the EST may produce a higher compaction in information. The eigenvalues approach zero after the 40th or so eigentarget, so we were interested in no more than the 40 most dominant eigentargets, instead of all 800 eigentargets. For setups a, b, and c, we used the 1, 5, 10, 20, 30, and 40 most dominant eigentargets of each transformation to produce the projection values for the MLP. For setup d, we used the 1, 5, 10, 20, and 25 projection values of each band to feed the corresponding MLPs with 2, 10, 20, 30, 40, and 50 input nodes, respectively. In each case, five independent training processes were tried with different initial MLP weights. The average hit rates of each setup for detector-centered chips, at a controlled false-alarm rate of 3%, are tabulated in Table 1. The bold numbers in the table indicate the best PCA and EST performance achieved for each setup with this dataset.

Table 1 Performance on Detector-Centered Chips at 3% False-Alarm Rate

No. of MLP inputs[a]	Data type	Average hit rates of five runs (%)							
		a		b		c		d	
		PCA	EST	PCA	EST	PCA	EST	PCA	EST
1/2	Train	21.08	46.31	26.31	43.96	25.41	48.83	25.55	50.05
	Test	20.07	42.68	23.69	40.87	22.78	45.21	23.98	45.28
5/10	Train	78.02	79.10	72.93	78.02	82.84	85.05	87.14	85.48
	Test	70.27	70.78	61.05	65.50	74.40	77.21	76.20	74.07
10/20	Train	79.93	81.69	76.40	79.86	88.25	**90.59**	**88.22**	**90.20**
	Test	73.24	72.88	63.00	67.05	79.49	**81.88**	**78.66**	**74.14**
20/30	Train	**83.35**	**85.01**	**79.06**	**85.30**	89.69	89.04	85.66	87.57
	Test	**74.50**	**74.47**	**66.91**	**69.26**	81.66	76.17	77.87	74.29
30/40	Train	79.17	80.29	78.81	76.72	**91.78**	85.55	80.94	88.32
	Test	66.91	64.34	66.76	61.05	**80.25**	71.86	73.27	72.19
40/50	Train	68.18	57.48	70.09	62.25	88.50	82.63	74.67	76.14
	Test	62.82	48.35	62.17	51.97	78.70	68.54	70.38	65.06

[a] First number is for setups a, b, and c. Second number is for setup d.

Comparing setup a and b in Table 1, we can see that the MW band performed better than the LW band when a moderate number of 5–30 projection values were fed to the MLP. For both setups, the peak performance was achieved with 20 MLP inputs. Although their peak hit rates for the training set are somewhat comparable, the MW leads in the testing performance by 5–8%. Therefore, the MW sensor seems to be the better candidate than the LW, if we have to choose only one of them for our clutter rejector. It should be noted that this conclusion may apply only to the specific sensors used for this study. If we compare setup a with setup c, we note significant improvement achieved by the stacked dual-band input in both training and testing sets, which ranges from 5% to 8% again. In other words, processing the MW and LW jointly is better than using either one of them alone. The way we merge the two bands also affects the clutter rejection performance. Although the performances of setups c and d are similar, setup c is the clear winner when it comes to the peak performance and in the cases where 20 or more MLP inputs were used. Therefore, combining the dual band before the eigenspace transformation, rather than after, is the better way to utilize the MW and LW jointly.

In order to examine the effect on the clutter rejector of accurate centering of the targets within the input chips, we repeated the above experiments with the second dataset. Once again, we tabulated the average hit rates achieved by each setup in Table 2 and marked with bold numbers the best performance of all setups. When we look at the best performance in Table 2, the relationships among the four setups are similar to those exhibited in Table 1. Due to the distinctly improved target chips in this case, performance of all setups have dramatically improved. Emerging from much lower hit rates on the first dataset, the single-band setups have made a greater gain than the dual-band setups with the improved target centering offered by the second dataset. As a result, the performance edge of the dual-band clutter rejectors has shrunk to about 5%. In other words, the usefulness of dual-band input would be reduced if the prior target detector could detect the ground-truth target center more accurately.

Finally, we repeated the same set of experiments on the third dataset, in which the target chips were centered and zoomed correctly using the ground-truth information. We give the average hit rates of each setup in Table 3. With a quick glance on the bold numbers in Table 3, one can see that near-perfect hit rates were achieved by almost every setup for the training set, even at a demanding 3% false-alarm rate. The performance on the testing set are not far behind either, with those of the setup a tailing at around 94%. In other words, accurate zooming of the target has helped every setup, especially the weaker single-band clutter rejectors.

Table 2 Performance on Ground-Truth-Centered Chips at 3% False-Alarm Rate

No. of MLP inputs[a]	Data type	Average hit rates of five runs (%)							
		a		b		c		d	
		PCA	EST	PCA	EST	PCA	EST	PCA	EST
1/2	Train	26.50	45.16	35.24	48.64	31.01	50.72	34.14	56.63
	Test	27.37	47.26	35.57	48.26	29.95	51.74	34.18	56.86
5/10	Train	89.92	89.93	87.44	85.41	92.31	94.34	94.00	**95.38**
	Test	85.92	83.83	85.42	85.42	90.25	90.85	88.71	**91.14**
10/20	Train	**92.11**	**93.40**	**91.02**	**88.88**	94.84	96.58	**97.87**	93.60
	Test	**85.27**	**85.07**	**86.81**	**86.37**	88.26	89.35	**89.40**	87.21
20/30	Train	90.47	88.69	83.47	80.00	**97.47**	**97.37**	95.43	95.39
	Test	86.97	79.31	80.20	73.73	**91.29**	**90.94**	89.80	87.31
30/40	Train	71.96	67.10	77.02	66.70	97.96	92.11	87.84	89.83
	Test	71.69	62.84	71.14	60.60	89.65	86.82	84.83	81.15
40/50	Train	77.92	70.67	79.30	69.53	82.08	84.96	87.59	73.10
	Test	75.57	62.64	73.58	64.93	81.14	80.65	84.83	66.07

[a] First number is for setups a, b, and c. Second number is for setup d.

Table 3 Performance on Ground-Truth-Centered-Zoomed Chips at 3% False-Alarm Rate

No. of MLP inputs[a]	Data type	Average hit rates of five runs (%)							
		a		b		c		d	
		PCA	EST	PCA	EST	PCA	EST	PCA	EST
1/2	Train	68.98	77.42	79.40	82.88	80.40	86.10	80.55	86.85
	Test	70.15	78.86	75.87	82.09	78.11	83.33	78.76	83.83
5/10	Train	70.22	97.17	78.86	**99.01**	77.32	**100.00**	80.35	**100.00**
	Test	71.49	95.62	80.55	**96.51**	79.15	**98.46**	82.59	**97.11**
10/20	Train	83.97	**99.95**	87.10	96.43	92.36	99.55	93.20	90.08
	Test	88.65	**94.73**	90.55	94.48	95.82	96.92	96.32	89.16
20/30	Train	88.29	92.70	90.57	92.61	94.64	98.96	96.43	95.09
	Test	91.99	85.97	93.63	87.56	96.12	92.78	97.51	89.55
30/40	Train	90.42	93.50	93.45	84.77	99.06	92.31	99.20	99.01
	Test	92.19	82.04	95.52	86.47	95.07	89.50	96.37	89.55
40/50	Train	**96.77**	93.30	**100.00**	87.74	**100.00**	98.51	**99.30**	99.35
	Test	**94.58**	83.93	**96.47**	85.82	**98.36**	89.50	**97.66**	89.60

[a] First number is for setups a, b, and c. Second number is for setup d.

In Table 4, we show the average value of the bold numbers in Tables 1–3 for the single-band (columns 3–6) and dual-band (columns 7–10) setups, respectively. The benefit of dual-band data decreases gradually as more ground-truth information is added to the process of chip extraction. It should be noted that as the performance improves, the performance estimates become relatively less accurate because of reduced number of samples.

The average recognition rates usually increase with the number of eigenvectors used for feature extraction, but they approach saturation at around 20 projection values. Theoretically, the more eigenvectors employed in the transformation, the larger the amount of information that should be preserved in the transformed data. However, using more transformed inputs increases the complexity of the MLP, prolongs the training cycle, results in an overfitted MLP with reduced generalization capability, and increases the chance of getting stuck in a nonoptimal solution. In our experiments, many clutter rejectors with a large number of projection values have shown a steady decrease in their peak performance, mainly because of the weakening in their generalization capability to recognize the targets in the testing set. When fewer projection values are used, a higher performance is achieved by the EST. This improvement can be attributed to the better compaction of

Table 4 Performance Improvement (%) by Dual-Band Data at 3% False-Alarm Rate

Data type	Single band	Dual band	Improvement
Detector centered			
Train	83.43	90.20	6.77
Test	71.29	78.73	7.44
Ground-truth centered			
Train	91.35	97.02	5.67
Test	85.88	90.69	4.81
Ground-truth-centered zoomed			
Train	98.93	99.83	0.90
Test	95.57	97.90	2.33

information associated with EST. However, the PCA performed as good or even better when more projection values were used, which may indicate that some minor information might have been lost in the EST method. Nonetheless, the EST should be a better transformation when only a small number of projection values can be processed, because of speed or memory constraints.

We also investigated the effect on the performance of clutter rejectors of jointly optimizing the transformation layer with the BMLP. Consider the room for potential improvement at a 3% false-alarm rate; we chose the best PCA setups with 5 (10 for setup d) MLP inputs that were trained with the third dataset. First, we tried to minimize the overall output error of the PCAMLP by modifying the PCA eigenvectors, based on the errors back-propagated from the BMLP, using the supervised Qprop algorithm only. The clutter rejection rates of these four PCAMLPs for the first 4000 epochs of joint Qprop optimization are shown in Fig. 17. Due to the increased discriminability at the PCA transformation layer, their hit rates were improved by 15–25%. The improvements achieved by single-band setups were especially significant and, therefore, further diminished the dwindling advantage held by dual-band setups for this dataset. The best testing performance of setups a–d were achieved at epoch 5862, 5037, 1888, and 5942 of training, with corresponding hit rates of 99.78%, 100.00%, 97.99%, and 100.00% for the training set and 98.22%, 98.66%, 96.44%, and 99.78% for the testing set, respectively.

We also attempted to modify the PCA transformation layer with Eq. (20), where the Qprop and GHA were applied simultaneously. The resulting improvements of the same PCA setups are shown in Fig. 18. Comparing the corresponding curves in Figs. 17 and 18, we found that the GHA appeared

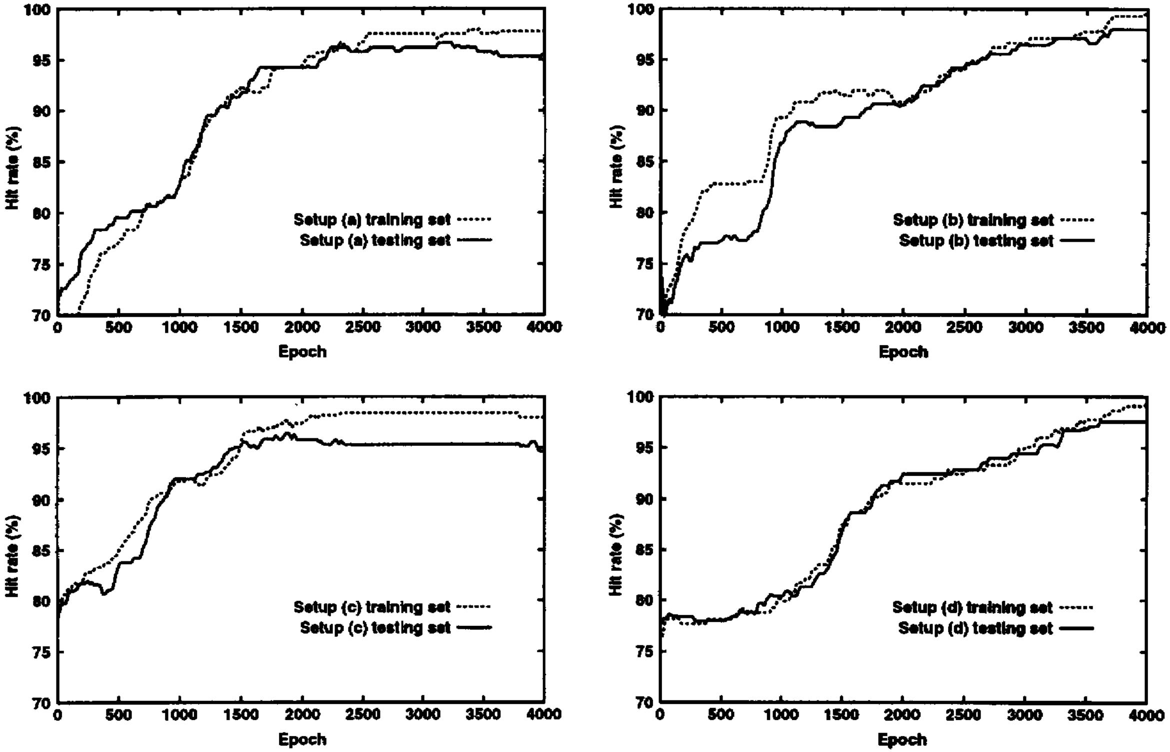

Figure 17 Clutter rejection performance of PCAMLP were enhanced by optimizing the PCA layer using the Qprop algorithm only.

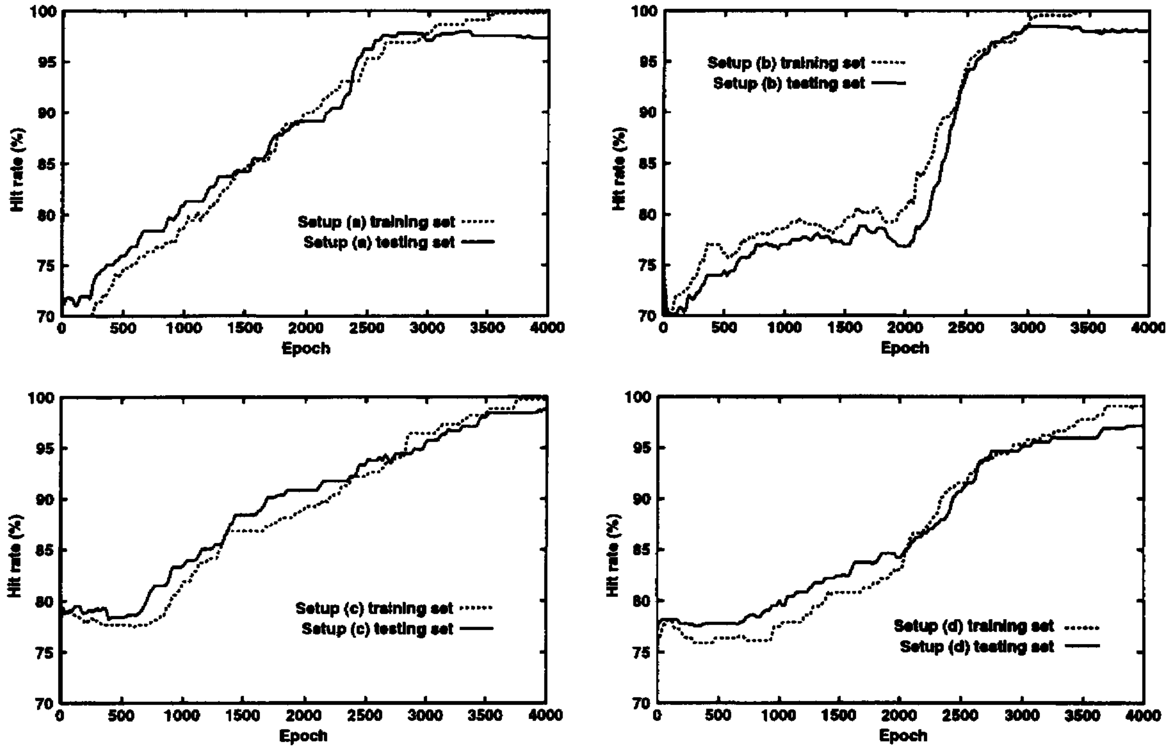

Figure 18 Clutter rejection performance of PCAMLP were enhanced by optimizing the PCA layer using Qprop and GHA algorithms simultaneously.

to slow down the improvement during the early stage of training, but then accelerated at the later stage to performance peaks that rival or beat those in Fig. 17. In this case, their best testing performance were achieved at epoch 3293, 3413, 3952, and 4531 of training, with corresponding hit rates of 99.11%, 100.00%, 99.78%, and 99.78% for the training set, and 98.00%, 98.44%, 99.33%, and 98.44% for the testing set, respectively. The early damping in learning curves indicates conflicting roles played by GHA and Qprop. The GHA tried to preserve the compaction characteristics of the transformation layer by maintaining the structures of those standard PCA eigenvectors, whereas the Qprop attempted to modify them in order to minimize the overall errors at the BMLP output node. The result of this struggle is a transformation layer that maintained most of its structure while emphasized some key areas, as exemplified by Fig. 12e.

Although the GHA did help the curves in Fig. 18 to reach their peaks sooner or higher, these differences in performance are statistically questionable because of the extremely small sample size. (The number of additional targets that are rejected by a system with 98.44% performance, versus 98.66% performance, is 1.) A larger or more difficult dataset is required to adequately measure the performance of this algorithm.

The added cost of computing the GHA is quite significant. Therefore, the usefulness of Eq. (20) is not proven by these experiments, where the transformation layer was initialized with standard PCA eigenvectors rather than random weights. In situations where the PCAMLP setups were equipped with the EST transformation layer, the effect of either joint optimization above was insignificant. The main reasons are thought to be associated with the integrated class separation formulation of the EST, as well as their near-perfect performance with merely five projection values.

1.4.2 PCAMLP as a Target Detector

The PCAMLP structure can be used as a target detector instead of a clutter rejector. As shown in Fig. 19, successive and overlapping chips can be extracted from the input frames and fed to the PCAMLP. For single-band detection, each chip is evaluated by the PCAMLP and the resulting output value indicates the likelihood of having a target situated at the location where the center of that chip is extracted. For setups c and d, a pair of chips must be extracted from the corresponding locations on the two bands for each evaluation. After the whole frame is evaluated, a number of locations with high PCAMLP scores are selected as potential target areas. High scores within a small neighborhood are combined and represented by the highest-scoring pixel among them. Any detection that lies sufficiently close to the ground-truth location is declared a hit, and if not, it is declared a

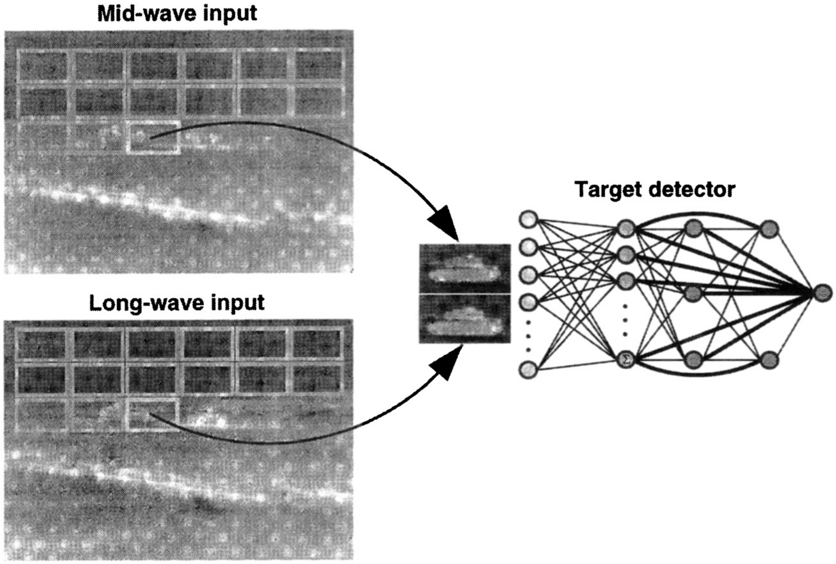

Figure 19 PCAMLP as a target detector.

false alarm. The numbers of hits and false alarms per frame could be changed by considering a different number of top detections from each frame.

We split the 461 pairs of LW–MW matching frames into two near-equal sets, each containing 286 targets of interest. We used the half with 231 frames as a training set, from which we extracted the training chips that were used in the previous clutter rejection experiments. In other words, the trained PCAMLP clutter rejections had "seen" parts of these frames, the parts where the NVDET detector declared as potential target areas. The other 230 served as a testing set, from which we extracted the testing chips for the clutter rejectors.

The same PCA setups chosen for the joint optimization experiments in Sections 1.4.1 were used as target detectors on these frames. With the standard PCA eigenvectors as their transformation layer, the detection performance of all four setups are presented as receiver operating characteristics (ROC) curves. The ROC curves obtained from the training and testing frames are shown at the upper and lower parts of Fig. 20 respectively. For the purpose of comparison, the ROC curves of the NVDET detector for MW and LW frames are also provided. Clearly, the single-band PCAMLPs outperformed the NVDET in both MW and LW cases at lower false-alarm rates, and the dual-band PCAMLPs excelled over the

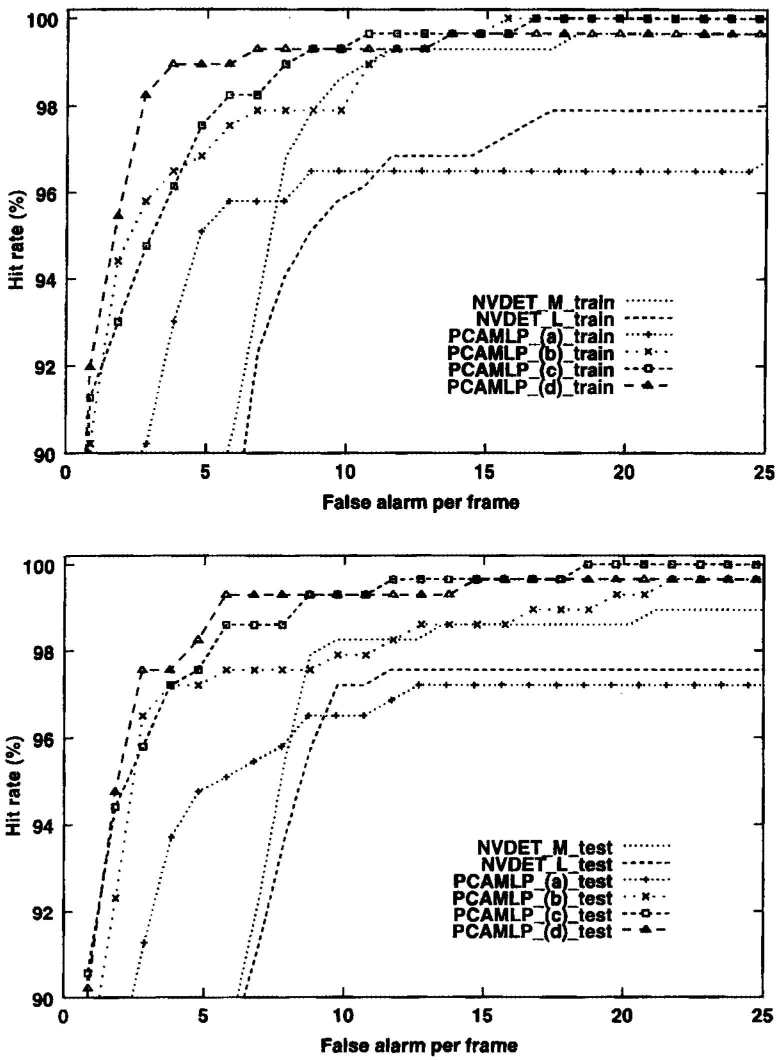

Figure 20 Detection performance of PCAMLP and NVDET.