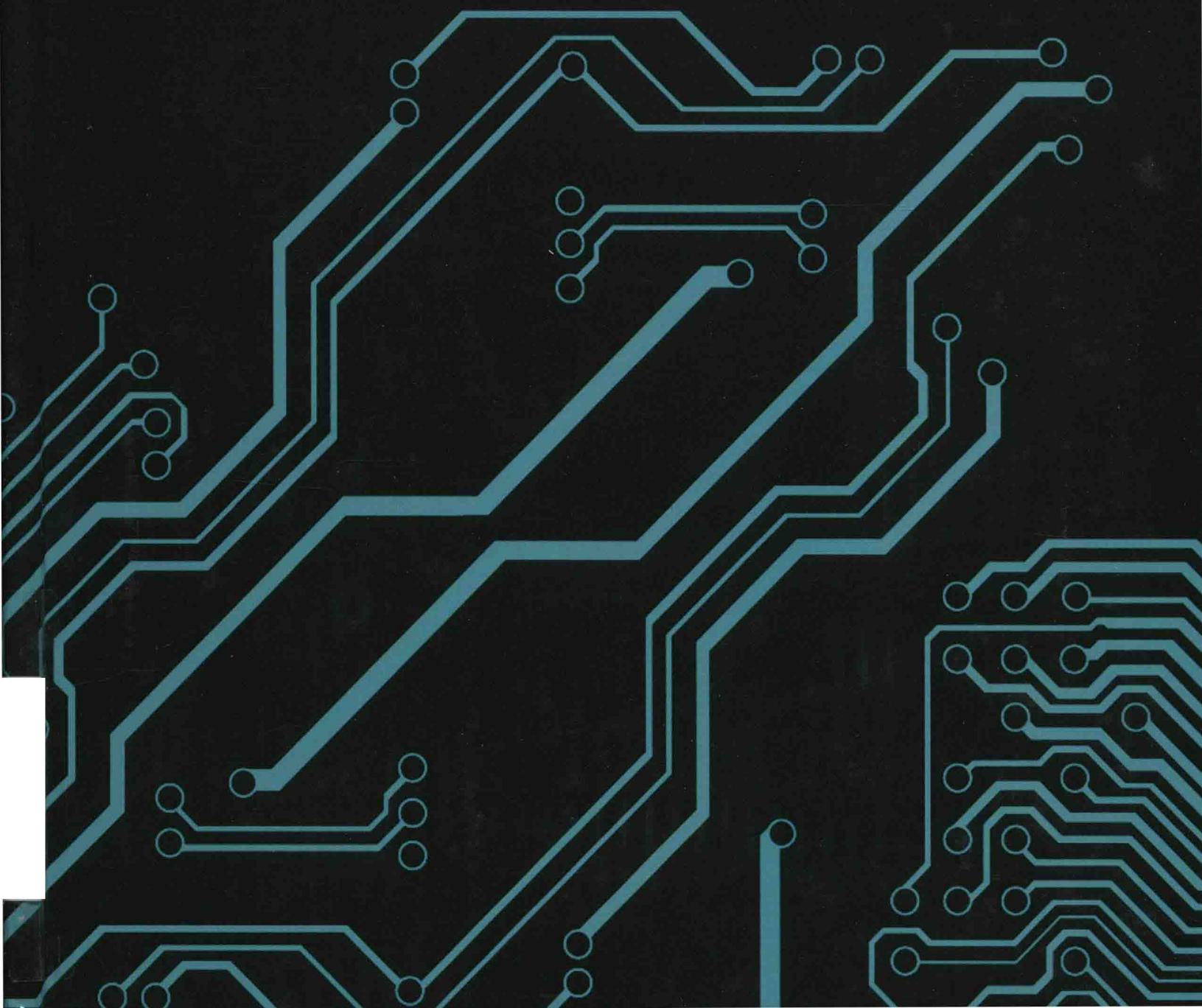
Handbook of Electrical and Computer Engineering

Volume I

Dewayne Hopper



1286799

Handbook of Electrical and Computer Engineering

Volume I

TN
H236H
V.1
2015

Edited by **Dewayne Hopper**

NY RESEARCH PRESS
New York

Published by NY Research Press,
23 West, 55th Street, Suite 816,
New York, NY 10019, USA
www.nyresearchpress.com

Handbook of Electrical and Computer Engineering: Volume I
Edited by Dewayne Hopper

© 2015 NY Research Press

International Standard Book Number: 978-1-63238-234-4 (Hardback)

This book contains information obtained from authentic and highly regarded sources. Copyright for all individual chapters remain with the respective authors as indicated. A wide variety of references are listed. Permission and sources are indicated; for detailed attributions, please refer to the permissions page. Reasonable efforts have been made to publish reliable data and information, but the authors, editors and publisher cannot assume any responsibility for the validity of all materials or the consequences of their use.

The publisher's policy is to use permanent paper from mills that operate a sustainable forestry policy. Furthermore, the publisher ensures that the text paper and cover boards used have met acceptable environmental accreditation standards.

Trademark Notice: Registered trademark of products or corporate names are used only for explanation and identification without intent to infringe.

Printed in China.

Handbook of Electrical and Computer Engineering

Volume I

试读结束，需要全本PDF请购买 www.ertongbook.com

Preface

The field of engineering that deals with the study and application of electronics and electricity is termed as electrical engineering. This field gained recognition as one of the biggest fields of engineering during the 19th century. Electric power has gained a foothold in the most basic activities of our lives. There is not a part of our daily lives that is not involved with electrical power in some way or the other. There are many subdivisions of electrical engineering like telecommunications, control systems, instrumentation and electronics, though electronics can be termed as a field in its own right. The skills required of an electrical engineer are variable and they work in a huge variety of industries. Computer engineering on the other hand is a discipline that combines various fields of electrical engineering and computer science and specializes in the development of computer hardware and software. Computer engineers can be involved in the many hardware and software facets of computing, like the design of microprocessors and personal computers to circuit design. It can be said that this field of study focuses on both how computer systems work as well as their integration in the larger picture. Both electrical and computer engineering are disciplines that can be counted as the biggest and fastest growing in the range of engineering fields that are there in the industry.

This book is an attempt to collate all current data and research on computer and electrical engineering. I am thankful to all the contributing authors for the hard work and effort put in these researches. I also wish to acknowledge the efforts of the publishing team who provided excellent technical assistance, whenever needed. Lastly, I wish to thank my friends and family who have supported me at every step in my life.

Editor

Contents

Evaluating Subpixel Target Detection Algorithms in Hyperspectral Imagery

Yuval Cohen,[1] Yitzhak August,[1] Dan G. Blumberg,[2] and Stanley R. Rotman[3]

[1] *The Unit of Electro-Optics Engineering and the Earth and Planetary Image Facility, Ben-Gurion University of the Negev, P.O. Box 653, 84105 Beer-Sheva, Israel*
[2] *The Department of Geography and Environmental Development and the Earth and Planetary Image Facility, Ben-Gurion University of the Negev, P.O. Box 653, 84105 Beer-Sheva, Israel*
[3] *Department of Electrical and Computer Engineering and the Earth and Planetary Image Facility, Ben-Gurion University of the Negev, P.O. Box 653, 84105 Beer-Sheva, Israel*

Correspondence should be addressed to Stanley R. Rotman, srotman@ee.bgu.ac.il

Academic Editor: James Theiler

Our goal in this work is to demonstrate that detectors behave differently for different images and targets and to propose a novel approach to proper detector selection. To choose the algorithm, we analyze image statistics, the target signature, and the target's physical size, but we do not need any type of ground truth. We demonstrate our ability to evaluate detectors and find the best settings for their free parameters by comparing our results using the following stochastic algorithms for target detection: the constrained energy minimization (CEM), generalized likelihood ratio test (GLRT), and adaptive coherence estimator (ACE) algorithms. We test our concepts by using the dataset and scoring methodology of the Rochester Institute of Technology (RIT) Target Detection Blind Test project. The results show that our concept correctly ranks algorithms for the particular images and targets including in the RIT dataset.

1. Introduction

Ideally, one would like to choose a hyperspectral detection algorithm for use in a particular scenario with the assurance that it would be "optimal," that is, that the type of algorithm to be used and its free parameters would be optimized for the particular task for which it is being considered. Of course, in such cases, the complexity of real-world scenarios and the difficulties of predicting the exact target signature *in situ*, make it hard to believe that we can predict the optimal target detection algorithm ahead of time. Because the responses of these algorithms can vary depending on target placement, we adapted the Rotman-Bar Tal Algorithm (RBTA) [1] for comparing point target detection algorithms, used for infrared broadband images, to the analysis of hyperspectral imagery [2–4]. The RBTA implants targets and evaluates the response of the detecting algorithm to their presence in every pixel in the dataset. Indeed, our development of new algorithms based on this tool has been validated by results obtained by other researchers in actual field tests [5, 6].

An inherent weakness of the RBTA method is its assumption that subpixel targets will each be contained within a single pixel. In light of our recent work [7], which showed that even very small targets can affect several pixels, here we fine-tuned the RBTA method to account for this possibility.

Sections 2–6 describes the RBTA in detail. We show how the simulation of target detection performance is dependent on the spatial correlation of the pixels present in the target.

Sections 7–12 analytically considers the expected performances of several detection algorithms under conditions of "pixel phasing," that is, a small target located simultaneously in several adjacent pixels. Our improved RBTA (IRBTA) takes into account target blurring and pixel phasing. The results presented in Sections 13–16 show that the superiority of the ACE algorithm and the importance of accounting for target blurring are validated in a real data analysis based on the

RIT target detection blind test experiment. Conclusions are presented in Section 17.

2. Determining the "Best Algorithm" for Target Detection

Manolakis et al. [8] claimed that to identify the best algorithm for target detection, we need datasets with reliable ground truth spanning a diverse range of targets and backgrounds under various scenarios, target fill factors, and atmospheric conditions. Statistically significant numbers of target and background pixels are necessary to construct reliable ROC curves. Because in many cases this degree of data confirmation is unavailable, we suggest an alternative approach for estimating the best algorithm from among several detectors for specific backgrounds and targets. We start by presenting the RBTA [1]. The algorithm was originally developed for broadband infrared images with subpixel targets, but we altered it to account for pixel blur (atmospheric and system effects which would cause the emitted power of the target to be spread over several pixels) and multipixel targets in hyperspectral imagery.

To estimate detector performance, Rotman and Bar-Tal proposed a multistep process that begins with an analysis of the unmodified reflectance image that is available in the website without any embedded targets. (We assume that ideally no targets are present in the datacube being analyzed; if one were present, it would slightly distort the histogram of the image. We trust that such a distortion will not disturb the overall analysis of the image statistics). The algorithm being tested is evaluated for each pixel, and the results are summarized in what we call a false-alarm histogram. Next we embed targets into every pixel and evaluate each of the algorithms. This is done independently for each pixel (rather than simultaneously) so that surrounding pixels are not changed prior to algorithm evaluation. The results are arranged in a target detection histogram. Each histogram (false-alarm and target detection) is then normalized; a variable threshold is set and the area of the false-alarm and target histograms to the right of the threshold are measured. For any particular threshold, a pair of P_{FA} and P_D (probability of false alarm and probability of detection) values are generated. The threshold is swept through all possible detector outputs, generating a set of these pairs. When graphed, these points produce the ROC (receiver operating characteristic) curves.

We note that the target implantation mechanism as given here has ignored several possibly significant effects which would affect the values found of PD. In particular, the target spectrum is a nearly noiseless lab spectrum that does not have the same artifacts, noise, and degradation as the real imagery. Additionally, this approach assumes the data has been perfectly atmospherically compensated by RIT's algorithm, which is not necessarily true. In our opinion, this seems to limit the use of our method rather than to invalidate it. Since the atmospheric conditions at the time of the measurement were not known, we cannot implant atmospherically corrected signatures or validate the reflectance dataset that is available in RIT website. Instead, we are testing the response of the algorithms to an implanted nonatmospherically corrected target which has been substituted in the reflectance dataset as described above; in each examined pixel, the fraction of the laboratory signature replaced the fraction of the background signal. While inaccurate atmospheric correction may result in an unknown decrease in the target detection, we note that the final comparisons are for variations in algorithm selection for a given target signature. The method should not be used to calculate absolute values for the probability of detection of a particular target which indeed has been altered by atmospheric and other effects. Rather, we are attempting to determine which algorithms will have a superior probability of detecting a target of this type in the scenario. Future work should include a quantitative determination to what degree atmospheric effects change the ranking of different algorithms.

This methodology can be used for the following reasons: as a rule, the ROC curve, which are generated tend to have probabilities of detection which range from 0 to 1; the value of probabilities of false alarm, on the other hand, vary from 0 to some chosen threshold $P_{\text{FA-MAX}}$. This threshold is normally set quote low; a standard value would be 0.01. This is appropriate since the acceptable use of most detection algorithms could only be in the range where a small percentage of the pixels in the image would be false alarms.

Now, the exact distribution of the background pixels is crucial for the analysis of our detection algorithms; it will indeed be the exceptional pixels in the tail of the distribution which will determine the ROC curve. However, since the probability of detection is being determined by the entire P_D scale from 0 to 1, all pixels contribute. In other words, the target detection scheme in this paper is extremely sensitive to a few false alarms; it is much less sensitive to a few pixels with missed "synthetic" target signatures. As such, subtle effects affecting the exact form of the target signature *in situ* are not being measured; rather the average response of the algorithm to the target signatures placement in all the pixels is the key factor. For our above goal, that is, the comparison of different target algorithms, we believe our method to be reliable.

To summarize, ROC curve evaluation entails the following steps as demonstrated in Figure 1.

3. Subpixel Target Detection: Global Methods

3.1. CEM. In many cases, it is convenient to scale the matched filter such that it has a value of 1 when the target signature fills the pixel being examined. This scaling can be achieved by normalizing the matched filter to its value when operating on the designated target spectrum:

$$\text{CEM}(\overline{x}) = \frac{\overline{s}^T \overline{\overline{R}}^{-1} \overline{x}}{\overline{s}^T \overline{\overline{R}}^{-1} \overline{s}}, \tag{1}$$

where s is the reference signature of the target, R is the background correlation matrix, that is, an [L × L] matrix, L is the number of bands, and x is the observed pixel. Geometrically speaking, the CEM algorithm measures the projection of x onto s normalized by the length of s in the

试读结束，需要全本PDF请购买 www.ertongbook.com

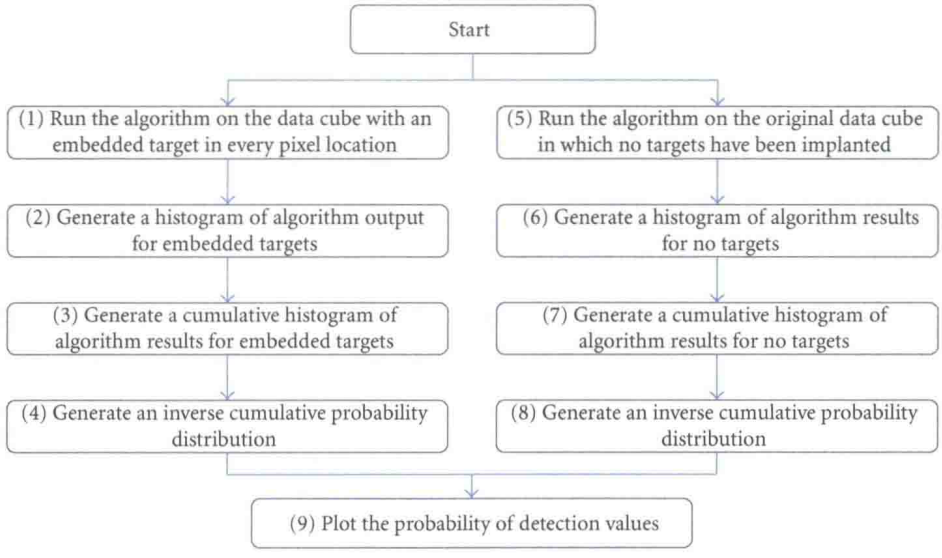


FIGURE 1: RBTA flow chart.

whitened space and thus leads to planar decision surfaces in that space. An important characteristic of the CEM algorithm is that its output is correlated to the target's fractional abundance in signature x, assuming the target signature is well isolated from the other endmembers, mixing is linear, and the relative abundances of the endmembers follow a Dirichlet distribution [9].

3.2. GLRT and ACE. Manolakis and his group [10–14] have described a number of stochastic target detection algorithms, including that attributed to Kelly [15] for solving to the Neyman-Pearson decision/detection theory for maximizing the probability of detection of a target with a fixed probability of false alarms. The solution uses a GLRT expressed as

$$\begin{aligned}\mathrm{GLRT}(\overline{x}) = &\left(\left[\left(\overline{s}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{x}-\overline{m_g}\right)\right]^2\right)\\ &/\left(\left[\left(\overline{s}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{s}-\overline{m_g}\right)\right]\right.\\ &\left.\cdot\left[1+(1/M)\cdot\left(\overline{x}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{x}-\overline{m_g}\right)\right]\right),\end{aligned} \tag{2}$$

where s and x are the same as for (1), m_g is the global mean, G is the background covariance matrix, and M is the total number of samples.

The ACE algorithm, a variation of the GLRT algorithm, is expressed as

$$\begin{aligned}\mathrm{ACE}(\overline{x}) = &\left(\left[\left(\overline{s}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{x}-\overline{m_g}\right)\right]^2\right)\\ &/\left(\left[\left(\overline{s}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{s}-\overline{m_g}\right)\right]\right.\\ &\left.\cdot\left[\left(\overline{x}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{x}-\overline{m_g}\right)\right]\right),\end{aligned} \tag{3}$$

with a maximum value of 1 for the case of $x = s$ and a minimum value of 0 when $x = m_g$.

In the context of target detection, the sign of $(s - m_g)^T G^{-1}(x - m_g)$ is important, as only positive abundances are of interest. (In contrast, this would not be the case for thermal gas detection, for example, where the target could be either absorptive or emissive in nature). Thus, in practice, a signed version of the GLRT algorithm is used as follows:

$$\mathrm{GLRT}_{\mathrm{sign}}(\overline{x}) = \mathrm{sign}\left[\left(\overline{s}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{x}-\overline{m_g}\right)\right]\cdot\mathrm{GLRT}(\overline{x}). \tag{4}$$

The corresponding ACE algorithm for target detection, also a variation of the GLRT algorithm, is expressed as

$$\mathrm{ACE}_{\mathrm{sign}}(\overline{x}) = \mathrm{sign}\left[\left(\overline{s}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}\left(\overline{x}-\overline{m_g}\right)\right]\cdot\mathrm{ACE}(\overline{x}). \tag{5}$$

Because real data does not necessarily match the assumptions from which the above algorithms are derived, that is, a background probability distribution function assumed to be multivariate Gaussian with zero mean bias and an additive target model, we generally cannot expect that any of the algorithms will be optimal or even that one will consistently outperform another [8]. Nevertheless, it was shown by Manolakis [13] that for a limited dataset, although each of the algorithms exhibited some degree of success in target detection, the ACE algorithm performed best on the limited dataset tested.

In Figure 1, step 1, note that the target is not in all the positions simultaneously; rather, the result is obtained sequentially. Steps 4 and 8 are generated by one minus the cumulative histogram using the results from step 3 and 7, respectively, (these are the probability of detection-PD). In Step 9, we plot PD values (step 4) versus the PFA (step 8).

4. Subpixel Target Detection Using Local Spatial Information

Improving target detection involved replacing the global mean with the local mean. Using the local mean is definitely double edged: on one hand, we would expect that the closer the points used to evaluate the background are to the suspected target, the more likely it is that the estimate will be accurate. On the other hand, the noise in the estimate will decrease given more points entering into the estimation, assuming that the background is stationary and the noise is linearly added to the background and independent thereof. Our empirical experience confirmed by several studies (4) and (5) is that the closer we choose the pixels the better, with the condition that we do not have target contamination of the background pixels. It is this proviso that we wish to test here.

We note that we are not dealing here with a "local" covariance matrix which would change when evaluating each pixel in the image. Rather, we use the same covariance matrix throughout the image; it will simply be based on the difference of the sample pixels and their "local" background.

Since we are dealing with a subpixel target, which in the physical domain can affect only pixels in a limited spatial area surrounding the center of the target, we used the eight nearest neighbors approach to estimate the value of the test pixels. The CEM algorithm does not use the mean and will therefore be unaffected by the above changes. The GLRT can be improved as follows:

$$\begin{aligned}\mathrm{GLRT}_{\mathrm{local}}(\overline{x}) = &\left(\left[\left(\overline{s}-\overline{m_g}\right)^T\overline{\overline{G}}^{-1}(\overline{x}-\overline{m_8})\right]^2\right)\\ &/\left(\left[(\overline{s}-\overline{m_8})^T\overline{\overline{G}}^{-1}(\overline{s}-\overline{m_8})\right]\right.\\ &\left.\cdot\left[1+(1/M)\cdot(\overline{x}-\overline{m_8})^T\overline{\overline{G}}^{-1}(\overline{x}-\overline{m_8})\right]\right),\end{aligned} \tag{6}$$

and for target detection

$$\begin{aligned}\mathrm{GLRT}_{\mathrm{sign-local}}(\overline{x}) = &\ \mathrm{sign}\left[(\overline{s}-\overline{m_8})^T\overline{\overline{G}}^{-1}(\overline{x}-\overline{m_8})\right]\\ &\cdot \mathrm{GLRT}_{\mathrm{local}}(\overline{x}),\end{aligned} \tag{7}$$

with m_8, the mean of the eight nearest neighbors, replacing the global mean m_g. For the ACE detector, the same procedure (replacing m_g by m_8) may be followed.

Segmentation [16–18] or even more local covariance matrices [2, 4, 6, 19] can be used to improve the covariance matrix. Common to all these methods is an increased need for high performance computational resources, while the corresponding influence each method has on detection ability is uncertain and highly dependent on the pictures being analyzed. Used in parallel, the algorithms create new difficulties through the combination of results from different segments. We used a global covariance matrix, but adapted it to local variations by using the local rather than the global mean, that is,

$$\begin{aligned}\overline{\overline{G}}_{\mathrm{global}} &= \frac{\left[\overline{\overline{X}}-\overline{\overline{M_g}}\right]^T\left[\overline{\overline{X}}-\overline{\overline{M_g}}\right]}{M},\\ \overline{\overline{G}}_{\mathrm{local}} &= \frac{\left[\overline{\overline{X}}-\overline{\overline{M_8}}\right]^T\left[\overline{\overline{X}}-\overline{\overline{M_8}}\right]}{M},\end{aligned} \tag{8}$$

where X is a two-dimensional matrix ($M \times L$), in which M is the number of pixels and L is the number of bands, m_g [$1 \times L$] is the mean vector of X, and M_g *is* m_g replicate M times. When we use M_8 for the covariance matrix, we do not need to replicate the mean, because M_8 is also of size [$M \times L$], and this is the appropriate covariance matrix for whitening $X - M_8$.

5. Data

We tested our algorithms on the online reflectance data sets and the hyperspectral data collected over Cooke City. The Cooke City imagery was acquired on 4 July 2006 using the HyMap VNIR/SWIR sensor with 126 spectral bands. Two hyperspectral scenes are provided with the dataset, one intended to be used for development and testing (the "Self Test" scene, where the positions of some targets are known) and the other intended to be used for detection performance evaluation (the "Blind Test" scene, where the position of targets is unknown). The data was corrected for atmospheric effects and available in the website but the exact atmospheric condition and the atmospheric correction algorithm are not available in the website and we assume that the reflectance dataset is good but not perfect. In Figure 2, we present the image in false color.

The target signatures, used both in the algorithm for detection and in the implantation of the synthetic targets in the RBTA method were laboratory measured and in reflectance units. The GSD is approximately 3 m. In Figure 3, we present the spectral signature of the targets in the blind test image.

The list of all targets is presented in Table 1 below.

6. Spatial Effect

6.1. Analytical and Simulated Performances of GLRT and ACE

6.1.1. Simple Case. The general form for local target detection as described in Section 3 is

$$\begin{aligned}D_{\mathrm{Local}}(\overline{x}) = &\left(\left[(\overline{s}-\overline{m_8})^T\overline{\overline{G}}^{-1}(\overline{x}-\overline{m_8})\right]^2\right)\\ &/\left(\left[(\overline{s}-\overline{m_8})^T\overline{\overline{G}}^{-1}(\overline{s}-\overline{m_8})\right]\right.\\ &\left.\cdot\left[\Psi_1+\Psi_2\cdot(\overline{x}-\overline{m_8})^T\overline{\overline{G}}^{-1}(\overline{x}-\overline{m_8})\right]\right),\end{aligned} \tag{9}$$

Table 1: Targets description.

Target ID	Target description	size (m^2) No. 1	size (m^2) No. 2	Self test ground truth	Blind test ground truth
F1	Red cotton fabric panel	3×3	N/A	Yes	No
F2	Yellow nylon fabric panel	3×3	N/A	Yes	No
F3	Blue cotton fabric panel	2×2	1×1	Yes	No
F4	Red nylon fabric panel	2×2	1×1	Yes	No
F5	Maroon nylon fabric panel	2×2	1×1	No	Web score
F6	Gray nylon fabric panel	2×2	1×1	No	Web score
F7	Green cotton fabric panel	2×2	1×1	No	Web score
V1	Chevy Blazer, green	4×2	N/A	Yes	Web score
V2	Toyota T100, white with black plastic liner	3×1.7	N/A	Yes	Web score
V3	Subaru GL Wagon, Red	4.5×1.6	N/A	Yes	Web score

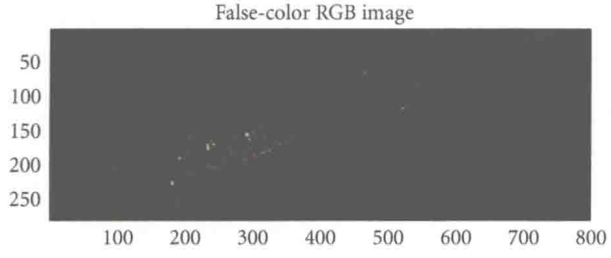


Figure 2: False-color RGB of the Cooke City imagery.

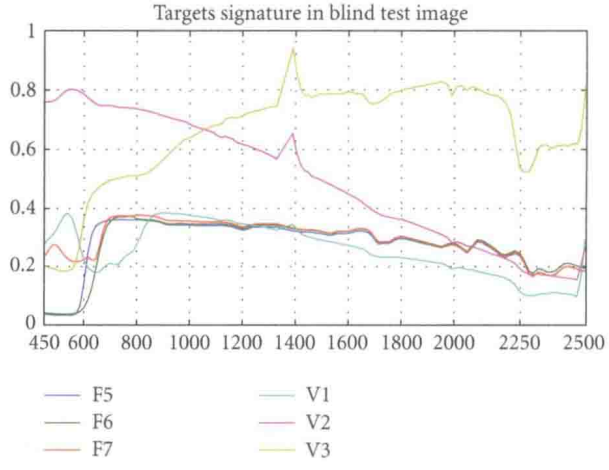


Figure 3: Spectral signatures of the targets that are present in the Blind test image x-axis is the wavelength [nm] and y-axis is the reflectance unit less.

with m_8 as the mean of eight neighbors. G, the global-local covariance matrix, is computed as

$$G_{\text{global\_local}} = \frac{\left(\overline{\overline{X}} - \overline{\overline{M_8}}\right)^T \cdot \left(\overline{\overline{X}} - \overline{\overline{M_8}}\right)}{L}, \tag{10}$$

where we can get GLRT and ACE as functions of $\Psi_1 + \Psi_2$:

$$\begin{aligned} &\text{GLRT} : \Psi_1 = M, \qquad \Psi_2 = 1, \\ &\text{ACE} : \Psi_1 = 0, \qquad \Psi_2 = 1. \end{aligned} \tag{11}$$

For the case in which the PUT (pixel under testing) x is exactly s, we obtain the following results:

$$\begin{aligned} D_{\text{Local}}(\overline{x}) = &\left(\left[(\overline{s} - \overline{m_8})^T \overline{\overline{G}}^{-1} (\overline{s} - \overline{m_8})\right]^2\right) \\ &/\left(\left[(\overline{s} - \overline{m_8})^T \overline{\overline{G}}^{-1} (\overline{s} - \overline{m_8})\right]\right. \\ &\left.\cdot\left[(\Psi_1 + \Psi_2) \cdot (\overline{s} - \overline{m_8})^T \overline{\overline{G}}^{-1} (\overline{s} - \overline{m_8})\right]\right). \end{aligned} \tag{12}$$

Let us define the scalar C as–

$$C = (\overline{s} - \overline{m_8})^T \overline{\overline{G}}^{-1} (\overline{s} - \overline{m_8}). \tag{13}$$

Therefore, when $\overline{x}$ is exactly $\overline{s}$, GLRT and ACE can be written as

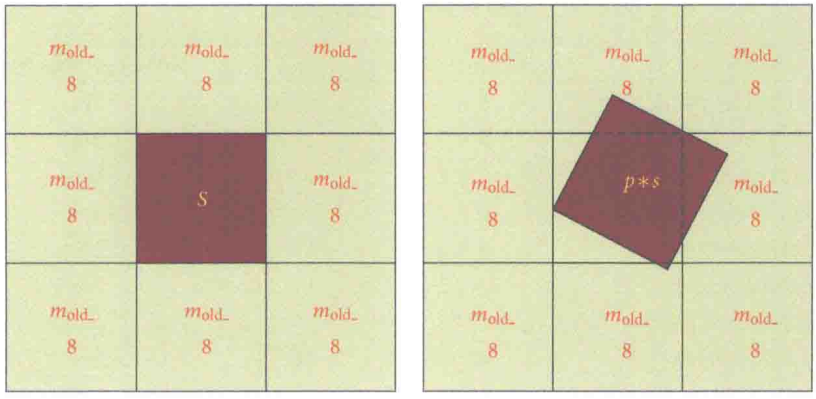
$$\begin{aligned} \text{GLRT}(\overline{s}) &= \frac{C}{[M + C]} = \frac{1}{M/C + 1}, \\ \text{ACE}(s) &= 1. \end{aligned} \tag{14}$$

Assuming that the data is normally distributed, C is chi-square distributed with $E(C) = \text{L}$, where L is the number of bands. For the case in which $M \gg E(C) = \text{L}$, we can assume that

$$\text{GLRT}(s) \cong \frac{C}{M}. \tag{15}$$

7. Pixel Phasing Case

When imaging, the target can often fall across several pixels even if its total size is only a single pixel; we will call this effect pixel phasing even though it is a natural consequence of imaging system quantization. The pixel phasing effect can be demonstrated by a target one pixel in size, the imaging of which leads to pixel phasing registration defined by p,



(a) Simple case (b) Pixel phasing case

FIGURE 4: Pixel phasing schema.

such that $0 < p < 1$ (0 corresponds to perfect sampling, with the target completely replacing the background) (Figure 4). From the point of view of the central pixel, it is not important the spatial location of the fraction within the pixel nor the location of the remainder of the target signature. Assuming uniform backgrounds of $m_{\text{old\_8}}$ for both center pixels, they can now be given as in Figure 4.

We obtain the following:

$$\overline{x_{\text{New}}} = p \cdot \bar{s} + (1 - p) \cdot \overline{m_{\text{old\_8}}}, \tag{16}$$

where x_{new} is the new PUT for the pixel phasing case and

$$m_{\text{new\_8}} = \frac{7 + p}{8} \cdot \overline{m_{\text{old\_8}}} + \frac{1 - p}{8} * \bar{s}, \tag{17}$$

where $m_{\text{new\_8}}$ is the new mean for the background.

We now evaluate the terms $(\bar{s} - \overline{m_{\text{new8}}})$ and $(\bar{x} - \overline{m_{\text{new8}}})$ as follows:

$$\begin{aligned}(\bar{s} - \overline{m_{\text{New8}}}) &= \bar{s} - \left[\frac{7 + p}{8} \cdot \overline{m_{\text{old\_8}}} + \frac{1 - p}{8} * \bar{s}\right] \\ &= \frac{(7 + p)}{8} \cdot (\bar{s} - \overline{m_{\text{old\_8}}}),\end{aligned} \tag{18}$$

$$\begin{aligned}(\overline{x_{\text{New}}} - \overline{m_{\text{New8}}}) &= (p \cdot \bar{s} + (1 - p) \cdot \overline{m_{\text{old\_8}}}) \\ &\quad - \left(\frac{7 + p}{8} \cdot \overline{m_{\text{old\_8}}} + \frac{1 - p}{8} * \bar{s}\right) \\ &= \frac{9 \cdot p - 1}{8} \cdot (\bar{s} - \overline{m_{\text{old\_8}}}).\end{aligned} \tag{19}$$

The GLRT result now becomes

$$\begin{aligned}D_{\text{Local}}(\bar{x}) = &\left([((9 \cdot p - 1)/8) \cdot (((7 + p)/8) \cdot C)]^2\right) \\ &/([((9 \cdot p - 1)/8) \cdot ((9 \cdot p - 1)/8) \cdot C] \\ &\cdot [\Psi_1 + \Psi_2((7 + p)/8) \cdot ((7 + p)/8) \cdot C]),\end{aligned} \tag{20}$$

where $D_{\text{Local}}(\bar{x})$ is the general local detector for the pixel phasing case.

For the case in which $N \gg C$, we calculate that

$$\begin{aligned}\text{GLRT}_{\text{miss\_sampling}}(\bar{x}) &\cong \frac{(p^2 + 14p + 49)}{64} \cdot \frac{C}{M} \\ &= \frac{(p^2 + 14p + 49)}{64} \cdot \text{GLRT}_{\text{local}},\end{aligned} \tag{21}$$

where $\text{GLRT}_{\text{miss\_sampling}}(x)$ is the expected GLRT value for the pixel phasing case and $M \gg L$. The GLRT expected value degrades as a function of p. But for ACE ($\Psi_1 = 0, \Psi_2 = 1$) we still get expected values of 1:

$$\text{ACE}_{\text{miss\_sampling}}(x) = 1 = \text{ACE}_{\text{local}}(x). \tag{22}$$

In this model, the complete lack of ACE degradation as a function of pixel phasing may explain why ACE is a more robust detector than GLRT in many test cases, as noted in the literature [8, 20].

8. Ranking the Algorithms by RBTA

The difficult task of synthesizing a synthetic image to help predict which algorithm to select is simplified and detector selection is facilitated if we synthesize only the target signature of our real image. Suppose we want to determine the proper detector for a specific target. We have already selected our method (e.g., CEM, GLRT, or ACE), and now we want to select the size of the local window. One approach is to assume that the best size for the local window is that under which the PUT value can be predicted with minimum error vis-à-vis the real PUT (in which we normalize each band by the mean values of the pixels in that band).

The approach outlined above depends only on the background image, not on the target signature, and it entails two assumptions: first, estimating signature values will improve our detector results independent of the different target signatures and second, the target has no effect on its neighbors. Address these assumptions in the following sections.

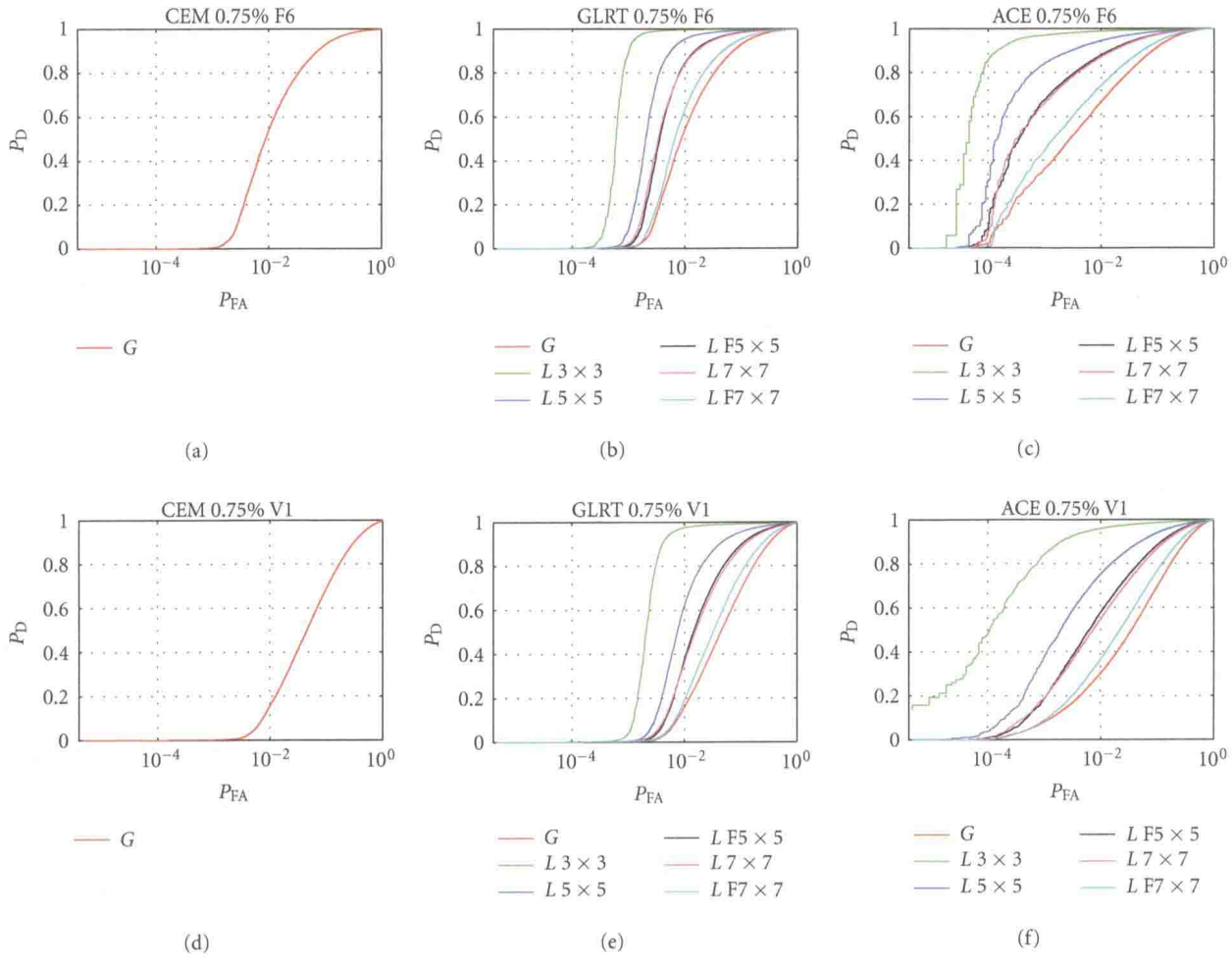


FIGURE 5: RBTA results for different size of local windows.

9. How to Use RBTA

The implementation of RBTA, which depends on our ability to implant realistic signals into backgrounds and measure detector response, should be done carefully. We cannot expect the real signature to be identical to a library signature, but we can hope for a high level of similarity. The low percentage of the target signature that actually enters any particular pixel is demonstrated in Figure 6; the response of the CEM filter, which responds proportionally to the percentage of the target fill in the tested filter, was maximum at 0.06.

As a rule, to test and challenge our algorithms by examining the area under the ROC curve, we need to test targets which neither "saturate" the ROC curve (with a probability of detection close to one with no false alarms detected) nor result in a "diagonal" ROC curve (in which the probability of detection equals the probability of false alarms. As the allowable false alarm rate decreases, the strength of our synthetic implanted target would need to increase; if we know what the acceptable false alarm rate is, we can select the target percent that will demonstrate the dynamic range around this rate and get results for our detectors (Figure 5).

For the values found experimentally for p, the target was easily detectable and saturated our ROC curve. Thus, we only embedded 0.0075 of the target signature in the background pixels to generate the target detection histogram. In our results, we found that for GLRT and ACE, the best local window size is 3×3 pixels (CEM has no local form). We also see that using bigger windows to estimate the pixel signature value gets us closer to the performance of global detectors that use a global mean. In this case, it is clea that local detectors are superior to global detectors.

In terms of real data, we must expect each target to affect more than one pixel even if its total physical size is at the subpixel level. A discussion of this point follows below and leads to improvement of the RBT algorithm.

10. Improvements to RBTA

10.1. Target Size. As will be discussed in Sections 11-12 the apparent target size in the final digital image is related both to its physical size and to various atmospheric and sensor effects, for example, its point spread function (PSF), Gibbs effect, crosstalk between pixels, spatial sampling, band-to-

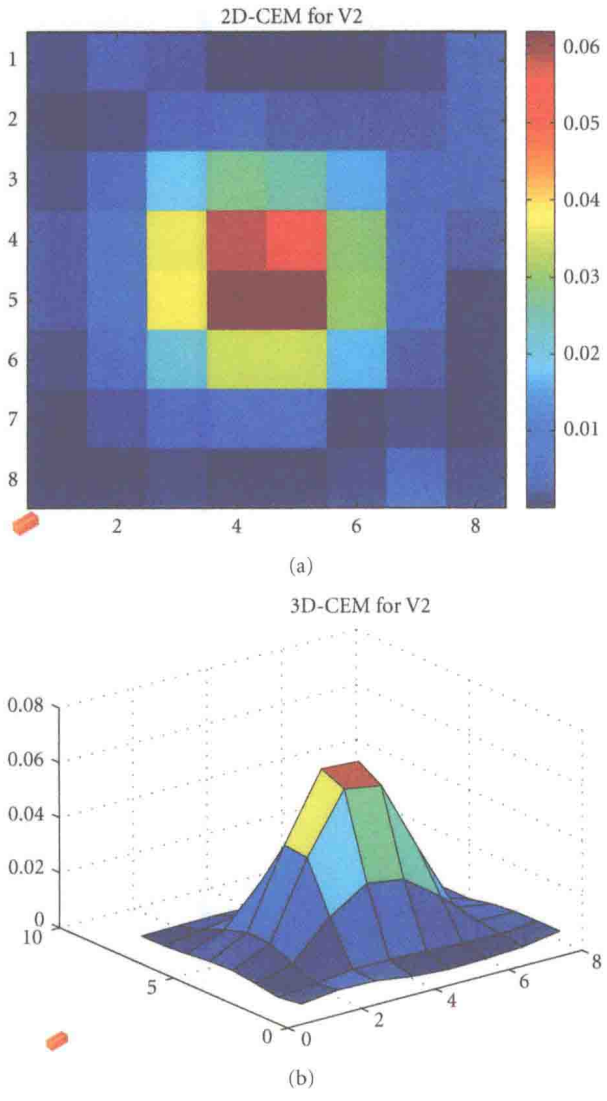


FIGURE 6: CEM results (2D and 3D) for target with pixel size.

band misregistration [5], and motion compensation. Thus, a target of a single pixel could actually occupy several pixels. In the RIT blind test, there are two 3×3-m targets, that is, exactly the size of the ground sample resolution (GSD) for the self and blind test images. Figure 6 shows a sample target of this size.

11. Spatial Sampling Effect

If we take into account only the spatial sampling effect, we can estimate the percent of pixel area partially occupied by the target. Notice that even targets of subpixel size often spread over neighboring pixels (Figure 7).

Put formally, the percent of pixel area covered as a function of target size, target location, and target orientation is

$$S_{\text{target\_in\_pixel}} = \iint_{-0.5}^{0.5} \frac{4}{\pi} \cdot (\alpha \cdot \beta) \cdot \text{UnitBox}[\tilde{x}, \tilde{y}] \cdot dx\, dy,$$

$$\tilde{x} = [\alpha \cdot [(x - \Delta x) \cdot \cos(\theta) - (y - \Delta y) \cdot \sin(\theta)]],$$
$$\tilde{y} = [\beta \cdot [(x - \Delta x) \cdot \sin(\theta) + (y - \Delta y) \cdot \cos(\theta)]], \tag{23}$$

where θ represents the clockwise rotation of the target relative to the pixel grid, and α, β represent the proportions of target length and width, respectively, relative to pixel physical dimension, Δx, Δy are the transition of the target origin relative to the pixel origin, as demonstrated in Figure 8

$$\text{UnitBox}[x, y] = \begin{cases} 1 & -\frac{1}{2} < x < \frac{1}{2} \text{ and, } -\frac{1}{2} < y < \frac{1}{2}, \\ 0 & \text{else.} \end{cases} \tag{24}$$

The expected value $E[S\alpha, \beta]$ is

$$E\left[S_{\alpha,\beta}\right] = \iiint_D S_{\text{target\_in\_pixel}}\, d\Delta x \cdot d\Delta y \cdot d\theta,$$
$$D = \begin{Bmatrix} 0 \le \Delta x \le 0.5 \\ 0 \le \Delta y \le 0.5 \\ 0 \le \theta \le \pi \end{Bmatrix}. \tag{25}$$

If we set θ to a constant value of zero and the target length and width are half the size of a pixel, we can simulate all the locations of the target where it's covering the same percent of pixel area (Figure 9).

Calculating (25) for a different size target using a numerical example produced the results shown in Figure 10.

In the graphs depicting pixel coverage as a function of physical target size, the x-axis is the ratio either between the target area and the pixel area (Figure 10(a)) or between the target length and the pixel length (Figure 10(b)). The blue line in both figures represents the percent of target within the pixel, while the green line is the percent of the pixel expected to be covered. It is intuitive that a very small target will be located in only one pixel, covering a small percent of that pixel. It is less intuitive, however, that the expected pixel to be covered will be entirely covered only by a target with an area four times that of the pixel.

12. Point Spread Function Effect

The PSF effect, present in any optical system, is not always known. Let us assume that the PSF is a typical, rotationally symmetric Gaussian filter of size 3 × 3 with standard deviation sigma 1/2.

Figure 11 demonstrates the synthetic spread effect that emerges from the convolution of the optical PSF (Figure 11(a)), and the physical pixels phasing due to target size (Figure 11(b)). For the spatial sampling we took the mean case representing the average pixel phasing that we could expect. Figure 11(c) represents the total effect, for example, convolution between (a) and (b). We devised an improved RBTA (IRBTA) and embedded the pixel signature and its neighbors with ratios as shown in Figure 11(c).

A comparison of Figures 5 and 12 shows that global detector and local detectors that both use only 7 × 7 frames

试读结束，需要全本PDF请购买 www.ertongbook.com