

Symmetrized Regression for Hyperspectral Background Estimation

James Theiler

Los Alamos National Laboratory, Los Alamos, NM 87545

ABSTRACT

We can improve the detection of targets and anomalies in a cluttered background by more effectively estimating that background. With a good estimate of what the target-free radiance or reflectance *ought to be* at a pixel, we have a point of comparison with what the measured value of that pixel *actually happens to be*. It is common to make this estimate using the mean of pixels in an annulus around the pixel of interest. But there is more information in the annulus than this mean value, and one can derive more general estimators than just the mean. The derivation pursued here is based on multivariate regression of the central pixel against the pixels in the surrounding annulus. This can be done on a band-by-band basis, or with multiple bands simultaneously.

For overhead remote sensing imagery with square pixels, there is a natural eight-fold symmetry in the surrounding annulus, corresponding to reflection and right angle rotation. We can use this symmetry to impose constraints on the estimator function, and we can use these constraints to reduce the number or regressor variables in the problem. This paper investigates the utility of regression generally – and a variety of different symmetric regression schemes particularly – for hyperspectral background estimation in the context of generic target detection.

Keywords: Background estimation, Target Detection, Anomaly Detection, Hyperspectral imagery

1. INTRODUCTION

To detect, identify, or quantify the properties of targets and/or materials of interest in a cluttered scene, a key step is the characterization of that background clutter. A number of standard detection algorithms – adaptive matched filter,¹⁻³ adaptive coherence estimator,^{4,5} finite target matched filter,⁶ *etc.* – begin with the assumption that the background is a multivariate Gaussian, whose mean μ and covariance R are estimated from the full dataset.

A natural way to improve this estimate is to compute the mean (and, possibly, the covariance) from a moving window centered on the pixel of interest. The venerable RX algorithm^{7,8} takes this approach for anomaly detection. The approach is less commonly employed for target detection, but recent efforts have illustrated the utility of local estimation for this purpose.⁹⁻¹³ A related approach^{14,15} employs an “unsharp mask” before applying a global detector; the effect of this filter is similar to the subtraction of a local mean.

Using the mean of the pixels in the annulus as the estimate of the central pixel is simple and effective, but possibly not optimal. In the regression framework, the central pixel is estimated from a general function of the pixels in the surrounding annulus. In one study,¹⁶ a k -nearest-neighbor algorithm (where neighbors are defined with respect to a “patch-space” of annulus-shaped patches) was used to obtain estimates with better SNR (*i.e.*, smaller residual variance) than was obtained using the mean of the pixels in the surrounding annulus.

Hasson *et al.*¹⁷ also employed a regression framework, and astutely observed that optimizing SNR does not necessarily optimize target detection performance. In particular, their variant of the patched-based nearest neighbor regressor achieved lower overall SNR than a mean or median for two small hyperspectral datasets, but it was the least effective of the three at the target detection tasks that were investigated. (One potentially important difference between the algorithms in Refs. [16] and [17] is that nearest neighbors are computed band-by-band in [16] but with all bands together in [17].)

It bears remarking that the problem of estimating a central pixel, based on the specific pixels that immediately surround it, along with the statistical context of the rest of the image, can be interpreted as a special case of the “inpainting” problem.¹⁸⁻²⁰ Here we are inpainting only a single pixel (or possibly a small patch, if we want to employ a “guard ring” between the candidate target location and the surrounding annulus), but we are doing it for every pixel in the image.

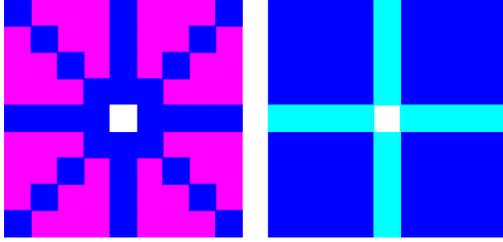


Figure 1: Left: Under the dihedral D_4 symmetry group, which permits reflections and rotations by right angles, some pixels (blue) have four-fold symmetry, and some (magenta) have eight-fold symmetry. Right: Under the more restricted Klein K_4 symmetry group, which permits only reflections and rotations by 180 degrees, most pixels (blue) have four-fold symmetry but some (cyan) have only two-fold symmetry.

2. REGRESSION FRAMEWORK

If y is the value of a pixel of interest, and $\mathbf{x} = [x_1, \dots, x_K]^T$ represents the pixels in the surrounding annulus, then we seek a function f for which $\hat{y} = f(\mathbf{x})$ is a good approximation for y . A simple first approximation uses the mean of the annular pixels: $f(\mathbf{x}) = (x_1 + \dots + x_K)/K$. But why not the median? Or how about a mean that puts extra weight on the pixels that are spatially closer to the central pixel? For that matter, one could consider an arbitrary linear combination: $f(\mathbf{x}) = a_1x_1 + \dots + a_Kx_K$.

In general, given a functional form for f , it is straightforward to learn the coefficients, because there are so many training examples. For every pixel (except those at the border of the image), we have a y value, and a surrounding annulus of \mathbf{x} values. For the linear model, for instance, we can just use least squares:

$$a_1, \dots, a_K = \operatorname{argmin}_a \sum_n (y_n - [a_1x_{1n} + \dots + a_Kx_{Kn}])^2 \quad (1)$$

where y_n is the value of the n 'th pixel, and x_{kn} is the k 'th pixel in the annulus surrounding the n 'th pixel.

For the experiments in this paper, only linear regression will be used. Indeed, one of the lessons of this work is that such a simple model can be very effective for characterizing the background of real multispectral and hyperspectral images. Some of the proposed regression algorithms (in particular, the $\Sigma\Delta$ scheme described in Section 3.4) will consider fixed nonlinear functions of the pixels (which can be thought of as “features” of the annulus), and linear regression will be applied to those features. That is,

$$\hat{y} = f(\mathbf{x}) = \sum_k a_k \phi_k(\mathbf{x}) \quad (2)$$

where ϕ_k is a possibly nonlinear function of the pixels x_1, \dots, x_K .

3. SYMMETRIES

Implicit in the analysis of imagery is the assumption that we can learn about the statistics of background variation in one part of the image by observing that variation in another part of the image. In particular, we expect the properties of background variability to be invariant to translation, rotation, and reflection. These are not strict symmetries: images are known to have nonstationarities, special alignments (*e.g.*, of streets in a city, or due to BRDF [bidirectional reflectance distribution function] effects when the sun is at a given azimuth), and (albeit rarely) specific “handedness” (*e.g.*, the direction of a spiral is not preserved under reflection). But even if they are occasionally broken, these symmetries are still useful approximations.

In general, we seek estimators $f(\mathbf{x})$ that are consistent with the symmetry operators. Thus, if g is a symmetry operator (*e.g.*, a rotation by ninety degrees), then we want $f(\mathbf{x}) = f(g\mathbf{x})$, where $g\mathbf{x}$ corresponds to the annulus pixels \mathbf{x} after operation by g . One way, we can enforce this behavior is by choosing features $\phi_k(\mathbf{x})$ that are symmetric; *i.e.*, $\phi_k(g\mathbf{x}) = \phi_k(\mathbf{x})$. It follows that the linear regression in Eq. (2) will be symmetric.

3.1. D_4 : the dihedral group

For an imager with square pixels* the natural symmetries are reflection and rotation by multiples of ninety degrees. These symmetries can be formally expressed in terms of the dihedral group D_4 , which contains eight

*If pixels were hexagonal, the symmetries would be richer.

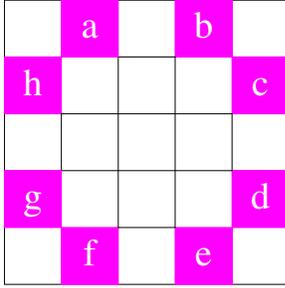


Figure 2: Eight pixels in a 5×5 array. Reflections and rotations by multiples of 90 degrees of the array will interchange the positions of these eight pixels. For instance, if s corresponds to reflections about the vertical axis, then $sa = b$, $sb = a$, $sc = h$, *etc.* If r corresponds to clockwise rotation by 90 degrees, then $ra = c$, $rb = d$, $rc = e$, *etc.*

elements. Each element is an operator, and the eight elements of the group can be generated with combinations of two generators, s and r , corresponding respectively to reflection about an axis (which we will take to be the vertical axis just to be specific), and rotation by ninety degrees (which we will take to be clockwise, again just to be specific). The group operation is successive application of the operators, with the usual convention that operators are applied in order from right to left; *e.g.*, the element rs corresponds to reflection (s) followed by ninety-degree rotation (r). Observe that the operators do not commute: $rs \neq sr$ (reflection followed by rotation is not the same thing as rotation followed by reflection).

We can specify the full group: $D_4 = \{1, r, r^2, r^3, s, sr, sr^2, sr^3\}$. The composition rules are: $r^4 = 1$ (four rotations by ninety degrees leaves the image unchanged); $s^2 = 1$ (two reflections revert back to the original image); and $rs = sr^3$ (reflection followed by a 90° rotation is equivalent to rotation by 270° followed by reflection). We can use these composition rules to multiply arbitrary group elements: *e.g.*, $r^2 \times sr^2 = r(rs)r^2 = r(sr^3)r^2 = (rs)r^5 = (rs)r = (sr^3)r = sr^4 = s$.

In general, if a pixel is at position (i, j) relative to a center pixel of interest, then the reflected pixel is at position $(-i, j)$ and the rotated pixel is at position $(j, -i)$. Applying all the elements of D_4 we get pixels $\{(\pm i, \pm j), (\pm j, \pm i)\}$ which in general is eight positions, but will only be four if either $i = 0$ or $j = 0$ or $i = j$. The singular case $i = j = 0$ corresponds the pixel in the center of the tile; this is seen in Fig. 1.

Consider the eight pixels - a, b, c, d, e, f, g, h - shown in Fig. 2. Group operations on these pixels will interchange the positions of the pixels, but the idea of this symmetry is that any function of these pixels that predicts the center pixel should be invariant to those re-arrangements. The simple mean function, $f(a, b, c, d, e, f, g, h) = (a + b + c + d + e + f + g + h)/8$, is trivially invariant to these re-arrangements, but it is possible to devise more complicated functions that are also invariant.

The regression idea is that the mean function can be replaced with more complicated functions, but the class of functions that might be used is vast. The symmetry idea is that this class might be reduced somewhat by imposing conditions associated with the symmetry. Invariance to reflection, for instance, is the statement that

$$f(a, b, c, d, e, f, g, h) = f(sa, sb, sc, sd, se, sf, sg, sh) = f(b, a, h, g, f, e, d, c) \quad (3)$$

Invariance to clockwise rotation by ninety degrees implies

$$f(a, b, c, d, e, f, g, h) = f(ra, rb, rc, rd, re, rf, rg, rh) = f(c, d, e, f, g, h, a, b) \quad (4)$$

If we can show these two invariances, then since s and r generate the full group, it follows that we have invariance to all the combinations of rotation and reflection that are in D_4 .

3.2. K_4 : the Klein group

For a pushbroom imager, one can imagine that the along-track noise and blur properties might differ from the cross-track properties, and so although we might want reflection (about both the along-track and cross-track axes), we might not want rotation by any angle other than 180 degrees. This leads to a four-fold symmetry, and the group $\{1, v, h, vh\}$, with $vh = hv$ corresponding to rotation by 180 degrees. Here $v^2 = h^2 = (vh)^2 = 1$, and this group is known as the Klein four-group (denoted K_4). It can also be described as $Z_2 \times Z_2$: the direct product of two copies of the order-2 cyclic group.

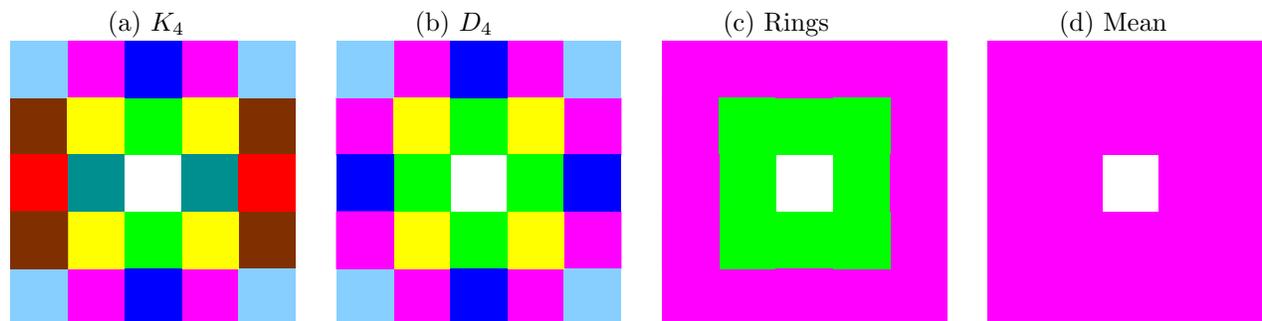


Figure 3: Coefficient patterns for various symmetries, shown for $(r_o, r_i) = (2, 1)$. Pixels with a common color in these annuli correspond to subsets that, for the linear Σ -only scheme, share a common coefficient. For the $\Sigma\Delta$ scheme, they correspond to subsets whose elements combine with each other (but not with other subsets) to produce the $\Sigma\Delta$ features. (a) Under the K_4 symmetry group, pixels with reflection symmetry share a common color. In this case, there are eight distinct subsets, which in the Σ -only case would correspond to eight total features. (b) The D_4 symmetry permits rotations as well as reflections, and there are fewer subsets (five in this case). (c) Here, the pixel subsets are concentric square rings. This arrangement does not lend itself to $\Sigma\Delta$ features, but for the linear Σ -only features, there are two coefficients for the $(r_o, r_i) = (2, 1)$ annulus. (d) With “Mean,” (as used by RX, for instance), all the pixels contribute equally.

Where the dihedral D_4 exhibits eight-fold symmetry, and generic pixels have seven counterparts (not including themselves), though some pixels have only three partners; the Klein K_4 group exhibits four-fold symmetry, and pixels generally have three counterparts, but some have only one partner. This is illustrated in Fig. 1, and in Fig. 3(a,b) in a different way.

3.3. Ring symmetry

One way to impose fuller rotational symmetry is to use full rings. That is, we surround a pixel with multiple concentric annuli, and permit a single coefficient for each annulus. Ref. [21] employed this scheme for background estimation in an astronomical context, and derived minimum-variance coefficients that would be consistent with polynomial interpolation. (A variant is obtained when the annuli contain Poisson counts.^{22, 23}) Here, we take a simpler approach, and use linear regression to identify the best coefficients.

With square pixels, we do not have the luxury of circular concentric annuli, but we can still make “square rings” of pixels for which $\max(|i|, |j|)$ is fixed.[†]

3.4. Sigma-Delta algebra

With the Sigma-Delta algebra, we perform a nonlinear transform on the data that enables us to maintain the imposed symmetry without reducing the number of features. Informally, we can think of these features as characterizing the “texture” of the pixel values in the annulus.

Let $\Sigma(A, B) = A + B$ be the sum, and $\Delta(A, B) = |A - B|$ be the absolute value of the difference. Note that Σ and Δ are both commutative with respect to their arguments; that is, $\Theta(A, B) = \Theta(B, A)$ for both $\Theta = \Sigma$ and $\Theta = \Delta$. In general, we can make a feature by combining:

$$\phi_{ZYX} = \Theta_X(\Theta_Y(\Theta_Z(a, b), \Theta_Z(e, f)), \Theta_Y(\Theta_Z(c, d), \Theta_Z(g, h))). \quad (5)$$

[†]Another alternative, not implemented here, is to make “diamond rings” of pixels for which $|i| + |j|$ is fixed.

In particular:

$$\phi_{\Sigma\Sigma\Sigma} = \Sigma(\Sigma(\Sigma(a, b), \Sigma(e, f)), \Sigma(\Sigma(c, d), \Sigma(g, h))) = a + b + e + f + c + d + g + h \quad (6)$$

$$\phi_{\Sigma\Sigma\Delta} = \Delta(\Sigma(\Sigma(a, b), \Sigma(e, f)), \Sigma(\Sigma(c, d), \Sigma(g, h))) = \left| (a + b + e + f) - (c + d + g + h) \right| \quad (7)$$

$$\phi_{\Sigma\Delta\Sigma} = \Sigma(\Delta(\Sigma(a, b), \Sigma(e, f)), \Delta(\Sigma(c, d), \Sigma(g, h))) = \left| (a + b) - (e + f) \right| + \left| (c + d) - (g + h) \right| \quad (8)$$

$$\phi_{\Sigma\Delta\Delta} = \Delta(\Delta(\Sigma(a, b), \Sigma(e, f)), \Delta(\Sigma(c, d), \Sigma(g, h))) = \left| \left| (a + b) - (e + f) \right| - \left| (c + d) - (g + h) \right| \right| \quad (9)$$

$$\phi_{\Delta\Sigma\Sigma} = \Sigma(\Sigma(\Delta(a, b), \Delta(e, f)), \Sigma(\Delta(c, d), \Delta(g, h))) = |a - b| + |e - f| + |c - d| + |g - h| \quad (10)$$

$$\phi_{\Delta\Sigma\Delta} = \Delta(\Sigma(\Delta(a, b), \Delta(e, f)), \Sigma(\Delta(c, d), \Delta(g, h))) = \left| (|a - b| + |e - f|) - (|c - d| + |g - h|) \right| \quad (11)$$

$$\phi_{\Delta\Delta\Sigma} = \Sigma(\Delta(\Delta(a, b), \Delta(e, f)), \Delta(\Delta(c, d), \Delta(g, h))) = \left| |a - b| - |e - f| \right| + \left| |c - d| - |g - h| \right| \quad (12)$$

$$\phi_{\Delta\Delta\Delta} = \Delta(\Delta(\Delta(a, b), \Delta(e, f)), \Delta(\Delta(c, d), \Delta(g, h))) = \left| \left| |a - b| - |e - f| \right| - \left| |c - d| - |g - h| \right| \right| \quad (13)$$

To see that these features are invariant to group elements, consider the effect of the two generating elements, s and r . Applying s to an arbitrary sigma-delta feature is seen to leave the feature unaltered:

$$s\phi_{ZYX} = \Theta_X(\Theta_Y(\Theta_Z(sa, sb), s\Theta_Z(se, sf)), \Theta_Y(\Theta_Z(sc, sd), s\Theta_Z(sg, sh))) \quad (14)$$

$$= \Theta_X(\Theta_Y(\Theta_Z(b, a), \Theta_Z(f, e)), \Theta_Y(\Theta_Z(h, g), \Theta_Z(d, c))) \quad (15)$$

$$= \Theta_X(\Theta_Y(\Theta_Z(a, b), \Theta_Z(e, f)), \Theta_Y(\Theta_Z(g, h), \Theta_Z(c, d))) \quad (16)$$

$$= \Theta_X(\Theta_Y(\Theta_Z(a, b), \Theta_Z(e, f)), \Theta_Y(\Theta_Z(c, d), \Theta_Z(g, h))) \quad (17)$$

$$= \phi_{ZYX}, \quad (18)$$

where the first step arose from applying s to each of $\{a, b, c, d, e, f, g, h\}$, and subsequent steps employed the commutative property of Θ_Z , followed by the commutative property of Θ_Y . Similarly,

$$r\phi_{ZYX} = \Theta_X(\Theta_Y(\Theta_Z(ra, rb), \Theta_Z(re, rf)), \Theta_Y(\Theta_Z(rc, rd), \Theta_Z(rg, rh))) \quad (19)$$

$$= \Theta_X(\Theta_Y(\Theta_Z(c, d), \Theta_Z(g, h)), \Theta_Y(\Theta_Z(e, f), \Theta_Z(a, b))) \quad (20)$$

$$= \Theta_X(\Theta_Y(\Theta_Z(c, d), \Theta_Z(g, h)), \Theta_Y(\Theta_Z(a, b), \Theta_Z(e, f))) \quad (21)$$

$$= \Theta_X(\Theta_Y(\Theta_Z(a, b), \Theta_Z(e, f)), \Theta_Y(\Theta_Z(c, d), \Theta_Z(g, h))) \quad (22)$$

$$= \phi_{ZYX} \quad (23)$$

and in this case, we made use of the commutative property of Θ_Y and Θ_X .

3.5. Sigma-only features

One property of $\Sigma\Delta$ features is that there are as many features as there were pixels to begin with. If we use only the Σ operator (which is linear), and not the Δ operator (which is nonlinear), then the eight features in Eq. (6) to Eq. (13), will be reduced to just the one in Eq. (6): $\phi_{\Sigma\Sigma\Sigma}$. For the $(r_o, r_i) = (2, 1)$ annulus, for example, the 24 pixels in the annulus are reduced to five features under the D_4 symmetry. The dimension reduction that this entails for other symmetries and annulus sizes is listed in Table 1.

4. REGRESSION AND INTERPOLATION

If we imagine that the spatial variability of the image can be modeled as a locally smooth function (or even locally smooth with noise), then we can interpret the estimation of the central pixel value as an interpolation. Indeed, this is the approach that was used to derive analytical expressions for estimating the background in an astronomical context.²¹

As a simple example, consider the $(r_o, r_i) = (1, 1)$ case, in which the annulus consists of the eight pixels immediately surrounding the pixel of interest. If we assume that the function $z(i, j)$ is a cubic polynomial in i and j , then we can derive the optimal coefficients for estimating the central pixel in terms of the annulus pixels.

Table 1: Number of features used, based on different symmetries, as a function of outer and inner radius of the annulus (r_o and r_i , respectively). The $\Sigma\Delta$ process does not change the number of features. The Σ -only symmetries reduce the number of features.

Symmetry	Number of features (r_o, r_i)					
	(1,1)	(2,1)	(3,1)	(3,2)	r_o, r_i	$r_o \gg r_i$
None	8	24	48	40	$(2r_o + 1)^2 - (2r_i - 1)^2$	$4r_o^2$
$K_4\Sigma\Delta$ (Klein)	8	24	48	40	$(2r_o + 1)^2 - (2r_i - 1)^2$	$4r_o^2$
$D_4\Sigma\Delta$ (Dihedral)	8	24	48	40	$(2r_o + 1)^2 - (2r_i - 1)^2$	$4r_o^2$
$K_4\Sigma$ (Klein)	3	8	15	12	$r_o^2 - r_i^2 + 2r_o + 1$	r_o^2
$D_4\Sigma$ (Dihedral)	2	5	9	7	$(r_o^2 - r_i^2 + 3r_o - r_i + 2)/2$	$r_o^2/2$
Diamond Rings	2	4	6	5	$2r_o - r_i + 1$	$2r_o$
Square Rings	1	2	3	2	$r_o - r_i + 1$	r_o
Mean	1	1	1	1	1	1

This optimal solution exhibits D_4 symmetry and can be expressed concisely as the weighted average of two averages: specifically,

$$\hat{z}(0, 0) = a_+ z_+ + a_\times z_\times \quad (24)$$

where

$$z_+ = [z(1, 0) + z(0, 1) + z(-1, 0) + z(0, -1)]/4 \quad (25)$$

$$z_\times = [z(-1, 1) + z(1, 1) + z(1, -1) + z(-1, -1)]/4. \quad (26)$$

Here z_+ corresponds to the average of the four pixels in the “cardinal” directions from the center, and z_\times is the average of the four corner pixels. Since this is a weighted average, we expect $a_+ + a_\times = 1$. We furthermore expect the cardinal pixels to have more weight than the corner pixels, since they are closer, and indeed they have twice as much weight. But this optimal solution is not $a_+ = 2/3$ and $a_\times = 1/3$. Instead, it is given by $a_+ = 2$ and $a_\times = -1$. That is, the four corner pixels not only have lower weight, their weight is negative.²⁴

This optimality result is altered if we incorporate noise. In particular, if the pixels are dominated by noise, then the best estimator is given by the simple average: $\hat{z}(0, 0) = 0.5z_+ + 0.5z_\times$. In practice, our images have some smoothness and some noise. And in that case, the best coefficients are somewhere “in between” $(2, -1)$ and $(0.5, 0.5)$. One can imagine making an explicit model of the image (*e.g.*, by a polynomial of a certain order along with noise at a certain level), followed by a derivation of the optimal coefficients to fit that model. But one can bypass the model and employ regression to learn those coefficients directly. Performing this experiment on a variety of images, it was empirically found that $(1.5, -0.5)$ was often close to the optimal solution.

5. EXTENSION TO MULTIPLE SPECTRAL CHANNELS

One can employ the methods described above to a multispectral or hyperspectral image by replacing scalars (pixel values at a given position) with vectors (pixel values over all wavelengths at a given position). This is the all-bands-together approach, and although it is conceptually straightforward, it can be computationally demanding, and otherwise problematic for nonlinear regression schemes.[‡]

To keep things simple, the experiments performed here employed a band-by-band approach. For the purposes of estimating the value of a central pixel inside an annulus of pixels, each band is treated separately. For a given exploitation task (*e.g.*, for target or anomaly detection), the bands are re-combined into a single image.

And for the experiments in this paper, this was done in two ways. In the *direct* band-by-band approach, the bands were just the usual bands associated with specific wavelengths. But it was often found that better

[‡]For example, it is hard to define an unambiguous “median” in multidimensional space.

Table 2: Performance measures for different choices of feature sets, as applied to different hyperspectral images. SNR is the signal-to-noise ratio, in decibels, that corresponds to the variance of the original image divided by the variance of the residuals; it is defined in Eq. (27). LVR is the log-volume-ratio corresponds to $\log(V_o/V)$, where V_o is the volume of the ellipsoid corresponding to the covariance of the original data, and V is the volume of the covariance ellipsoid of the residual data; it can be computed using the formula in Eq. (29). Higher values are better in both cases, although higher values of SNR are not *always* better.¹⁷

AVIRIS – Indian Pines						SpecTIR – Reno				HyMap – Cooke City			
Features	#	PCA		direct		PCA		direct		PCA		direct	
		SNR	LVR	SNR	LVR	SNR	LVR	SNR	LVR	SNR	LVR	SNR	LVR
Cardinal	1	9.0	-30.1	9.0	-30.1	7.8	-7.9	7.8	-7.9	11.4	85.2	11.4	85.2
Median	1	9.2	17.4	9.2	-21.0	9.4	26.4	9.4	4.9	12.5	98.1	12.5	3.1
Mean	1	8.6	20.9	8.6	20.9	8.8	26.8	8.8	26.8	12.1	99.8	12.1	99.8
Rings	2	11.8	33.3	11.8	11.9	13.9	38.8	13.9	29.0	22.5	266.0	22.4	264.6
$D_4\Sigma$	5	16.3	65.7	15.9	3.8	17.2	47.1	17.1	12.1	25.4	300.9	25.4	287.6
$D_4\Sigma\Delta$	24	16.3	65.8	16.0	2.2	17.3	47.0	17.3	9.4	25.5	300.8	25.5	283.7
$K_4\Sigma$	8	17.1	100.9	16.6	57.2	17.3	47.4	17.3	12.5	25.4	300.4	25.4	286.7
No-Sym	24	17.1	101.0	16.6	56.7	17.4	47.6	17.3	13.5	25.5	300.7	25.5	287.0

performance was achieved if the data were first transformed into principal components. The *PCA* band-by-band approach is applied to each principal component image, and then then the image is re-transformed back to usual wavelength bands.

It bears remarking that the baseline “mean” estimator – which estimates the central pixel as a simple arithmetic average of the pixels in the annulus – gives the same result in all three cases: all-bands-together, direct band-by-band, and PCA band-by-band.

6. RESULTS

This study considered three separate hyperspectral images: the classic AVIRIS^{25,26} image of Indian Pines, the HyMap blind test²⁷ image of Cooke City, and a SpecTIR image of Reno, Nevada.²⁸

Two sets of experiments were performed. The first involved estimating the background at each pixel and computing generic measures of accuracy for the different regression schemes. In the second set of experiments, more direct measures of detection performance were computed (ROC curves) by implanting anomalies into the scene. Although more specific target detection is often a more operationally useful task, the motivation for using non-specific anomalies was to allow us (indeed, to force us) to focus on the background estimation aspect of the problem.

6.1. Generic measures background estimation performance

Since the aim is to estimate the background, a very simple measure of estimation accuracy is the magnitude of the residuals. In particular, the SNR is defined by

$$\text{SNR}(\text{dB}) = -10 \log_{10} \langle \mathbf{e}^T \mathbf{e} \rangle + 10 \log_{10} \langle \mathbf{x}^T \mathbf{x} \rangle \quad (27)$$

where \mathbf{x} is the pixel in the original image, and \mathbf{e} is the pixel in the residual image. In general, a smaller residual indicates a larger SNR, which is usually (but not always¹⁷) better. Note that Eq. (27) can be expressed in terms of the eigenvalues of the original covariance matrix (denoted λ_i^o) and eigenvalues of the residual covariance matrix (denoted λ_i):

$$\text{SNR}(\text{dB}) = \frac{10}{\log 10} \left[\log \sum_i \lambda_i^o - \log \sum_i \lambda_i \right]. \quad (28)$$

Volume measures^{29–32} are inspired by a distribution based approach to anomaly detection. A full analysis would consider the volume of a distribution contour required to cover a given fraction (often 0.99 or 0.999) of the

residual pixel values, but we will treat the data as Gaussian, and use the determinant of the covariance matrix as our proxy for volume. (It is equal to volume times a fixed scale factor.) Rather than report an absolute value of the volume, however, we will take the ratio of the residual volume and the original volume. (Thus, the fixed scale factor cancels.) In particular, we will report the log volume ratio (LVR) given by $\log(V_o/V)$, which can be computed using

$$\text{LVR} = \log(V_o/V) = \sum_i \log \lambda_i^o - \sum_i \log \lambda_i. \quad (29)$$

Since smaller volumes are better, larger LVR's are preferred.

Table 2 show how well these estimators work in terms of both of these measures. A general trend is that as the number of features increases, the higher the SNR and LVR go. The most notable exceptions to this trend are in the direct band-by-band (*i.e.* *i.e.*, without PCA) columns of the LVR statistic. This speaks to the advantage of PCA rotation before band-by-band regression. Interestingly, the SNR statistic did not indicate much distinction between direct and PCA-based band-by-band regression. This is further evidence that the SNR statistic is limited in its usefulness for comparing these regression schemes. Another trend is that the non-symmetric regression, which has the highest number of features, did not substantially outperform the regression using many fewer symmetric features ($K_4\Sigma$ for Indian Pines and $D_4\Sigma$ for Reno and Cooke City).

6.2. Anomalous target detection performance

The ground truth problem (that it is often unreliable, and that there is never enough of it) is a perennial issue in algorithm development for remote sensing applications. One solution is to *implant* targets in the image.^{9,13,33–35} While this may not be quite as realistic as having actual targets in the actual scene, it has the advantage that the locations, spectra, and strengths of those targets are precisely known. One can further justify this approach by noting that for anomaly detection problems, the notion of a “realistic” target is difficult to define, anyway.³⁶ It also recognizes that a key (if not *the* key) component of any target or anomaly detection scheme in multi/hyperspectral imagery is the characterization of the cluttered background.^{37,38}

For the experiments here, we used uniform anomalies. For each band, we found the minimum and maximum value of the pixel in the image, and created target spectra whose values were band-by-band independent and uniformly distributed over minimum-to-maximum range. For each target \mathbf{t} that was generated this way, we mixed it into the scene at a random pixel location using a simple linear combination:

$$\mathbf{x}' = (1 - \alpha)\mathbf{t} + \alpha\mathbf{t} \quad (30)$$

Small values of α were used, which can be interpreted either as very small (subpixel) uniformly distributed anomalies, or else as full- (or nearly full-)pixel anomalous targets whose spectra are only subtly different from the normal background.

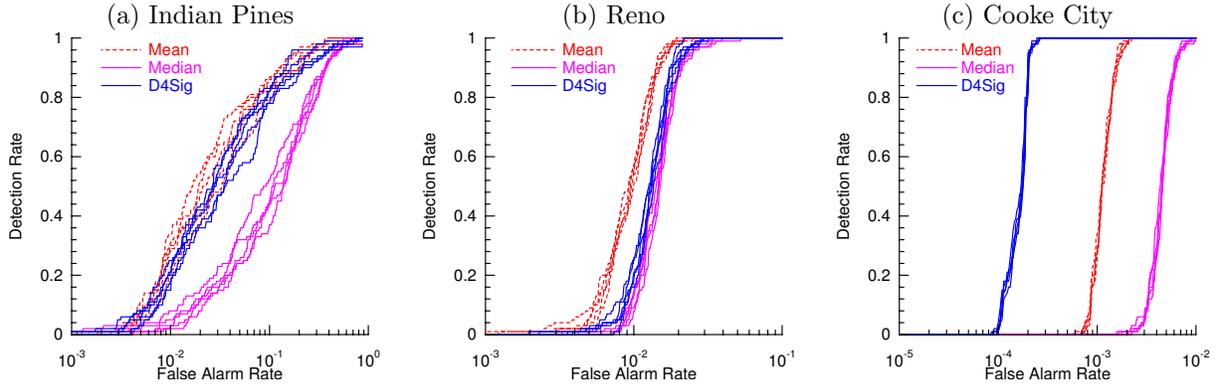
The anomaly detector used in these experiments is in the style of RX,^{7,8} but where RX uses a local mean of the annulus to estimate the center pixel, these detectors use one of the regression estimators. It has been noted that it is advantageous to compute covariance from a larger annulus than is used for estimating the mean.³⁷ These experiments take an extreme version of that advice, and compute the covariance globally, based on the residual image.

The ROC curves in Fig. 4 show how well this detector was able to identify the anomalies that were implanted into the images. The results seen in these curves more or less mirror the results in the LVR columns of Table 2. Specifically: PCA band-by-band is better than direct band-by-band; $D_4\Sigma$ is better then Mean, but only in the PCA band-by-band case.

7. CONCLUSIONS AND FUTURE WORK

The main result of this work is that the regression framework provides a potential for much better target detection than the traditional annulus mean. Further, in contrast to earlier attempts,^{16,17} this performance gain was obtained with computationally efficient linear regression. The use of symmetry did indeed enable competitive performance with fewer regressor variables, but the introduction of nonlinear symmetry-preserving

(a,b,c) Direct band-by-band



(d,e,f) PCA band-by-band

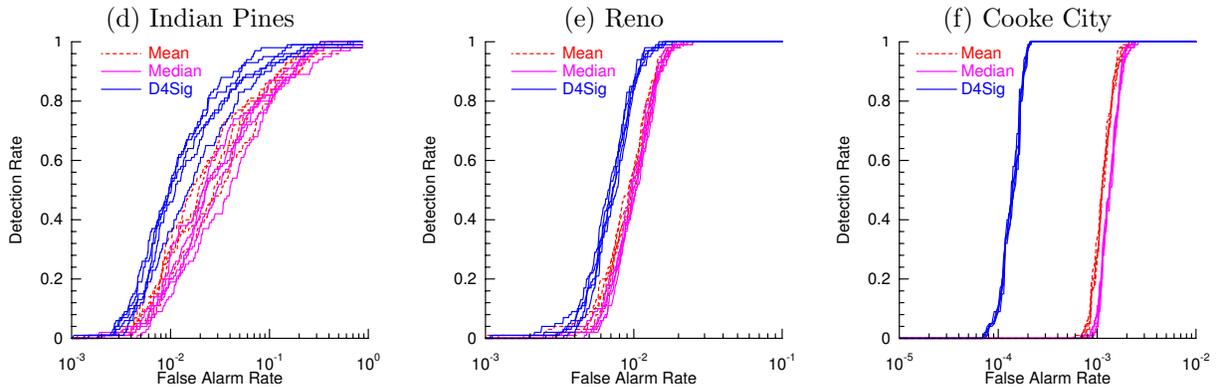
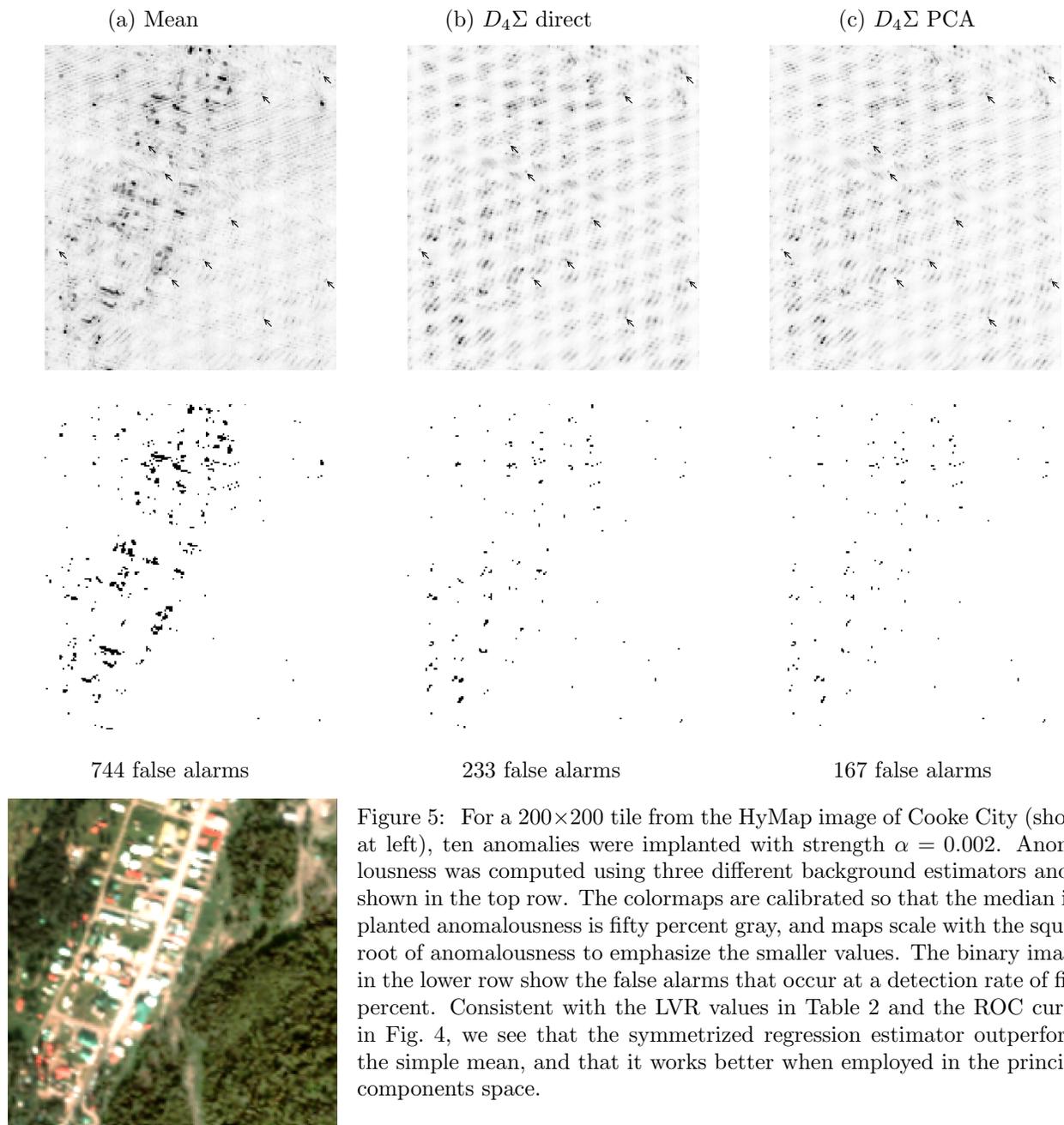


Figure 4: ROC curves illustrate performance of different algorithms for detecting anomalies that have been artificially implanted into the scene (at a level $\alpha = 0.01$ for Indian Pines and Reno, and $\alpha = 0.005$ for Cooke City). The experiment is repeated five times so that variability in the ROC curve can be observed. The top row is direct band-by-band background estimation; the bottom row uses principal components analysis. It is evident from this figure that PCA improves the performance of the median estimator (by a lot) and the $D_4\Sigma$ estimator (by a little), but has no influence on the mean (standard RX) estimator. The figure also shows that regression (in this case, using $D_4\Sigma$) outperforms the mean and median, in some cases by a substantial factor. Finally, we observe that the relative performance of the different anomaly detectors corresponds more directly with the LVR (log-volume-ratio) than with the SNR (signal-noise-ratio) statistics that are reported in Table 2.

$\Sigma\Delta$ features did not prove very helpful. It was found that PCA rotation was important for effective band-by-band regression, but it is not known if that is the best choice. Future work may consider other rotation schemes (such as PCA applied to the residuals instead of the original image).

At this point, the recommended background estimation algorithm is PCA-based band-by-band regression on the dihedrally symmetric $D_4\Sigma$ features. Future work may consider more overtly nonlinear regression methods, and for these, it is even more important to have a small number of features.

It bears remarking that this estimator is essentially a convolution with a fixed annular kernel, and there may be effective Fourier-domain interpretations and/or implementations that have not yet been pursued. More sophisticated spatial filtering³⁹ has shown promise in other image processing applications, and recent advances in efficient convolutional sparse coding^{40, 41} make this approach an attractive direction for further research.



ACKNOWLEDGMENTS

Brendt Wohlberg co-authored our original paper on the regression framework,¹⁶ and has offered insights on the present work that were numerous, detailed, and invaluable. Stanley Rotman and his colleagues have long championed the utility of local background modeling for target detection problems,^{10–13} and it was that work (and conversations with Stanley about that work) that led to this line of inquiry, and that continues to challenge¹⁷ and inspire it.

This work was supported by the United States Department of Energy (DOE) NA-22 Hyperspectral Advanced Research and Development (HARD) Solids project, and has benefited from discussions with the HARD Solids team.

REFERENCES

1. I. S. Reed, J. D. Mallett, and L. E. Brennan, “Rapid convergence rate in adaptive arrays,” *IEEE Trans. Aerospace and Electronic Systems* **10**, pp. 853–863, 1974.
2. E. J. Kelly, “An adaptive detection algorithm,” *IEEE Trans. Aerospace and Electronic Systems* **22**, pp. 115–127, 1986.
3. F. C. Robey, D. R. Fuhrmann, E. J. Kelly, and R. Nitzberg, “A CFAR adaptive matched filter detector,” *IEEE Trans. Aerospace and Electronic Systems* **28**, pp. 208–216, 1992.
4. L. L. Scharf and L. T. McWhorter, “Adaptive matched subspace detectors and adaptive coherence estimators,” *Proc. 30th Asilomar Conference on Signals, Systems, and Computers*, pp. 1114–1117, 1996.
5. S. Kraut, L. L. Scharf, and R. W. Butler, “The Adaptive Coherence Estimator: a uniformly most-powerful-invariant adaptive detection statistic,” *IEEE Trans. Signal Processing* **53**, pp. 427–438, 2005.
6. A. Schaum and A. Stocker, “Spectrally selective target detection,” *Proc. ISSSR (International Symposium on Spectral Sensing Research)*, p. 23, 1997.
7. I. S. Reed and X. Yu, “Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution,” *IEEE Trans. Acoustics, Speech, and Signal Processing* **38**, pp. 1760–1770, 1990.
8. A. D. Stocker, I. S. Reed, and X. Yu, “Multi-dimensional signal processing for electro-optical target detection,” *Proc. SPIE* **1305**, pp. 218–231, 1990.
9. Y. Cohen and S. R. Rotman, “Spatial-spectral filtering for the detection of point targets in multi- and hyperspectral data,” *Proc. SPIE* **5806**, pp. 47–55, 2005.
10. C. E. Cafer, M. S. Stefanou, E. D. Nelson, A. P. Rizzuto, O. Raviv, and S. R. Rotman, “Analysis of false alarm distributions in the development and evaluation of hyperspectral point target detection algorithms,” *Optical Engineering* **46**, p. 076402, 2007.
11. C. E. Cafer, J. Silverman, O. Orthal, D. Antonelli, Y. Sharoni, and S. R. Rotman, “Improved covariance matrices for point target detection in hyperspectral data,” *Optical Engineering* **47**, p. 076402, 2008.
12. Y. Cohen, D. G. Blumberg, and S. R. Rotman, “Subpixel hyperspectral target detection using local spectral and spatial information,” *J. Applied Remote Sensing* **6**, p. 063508, 2012.
13. Y. Cohen, Y. August, D. G. Blumberg, and S. R. Rotman, “Evaluating sub-pixel target detection algorithms in hyper-spectral imagery,” *J. Electrical and Computer Engineering* **2012**, p. 103286, 2012.
14. C. C. Borel and R. F. Tuttle, “Improving the detectability of small spectral targets through spatial filtering,” *Proc. SPIE* **7812**, p. 78120K, 2010.
15. C. C. Borel, “Methods to find sub-pixel targets in hyperspectral data,” *Proc. 3rd IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2011.
16. J. Theiler and B. Wohlberg, “Regression framework for background estimation in remote sensing imagery,” *Proc. 5th IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2013.
17. N. Hasson, S. Asulin, S. R. Rotman, and D. Blumberg, “Evaluating backgrounds for subpixel target detection: when closer isn’t better,” *Proc. SPIE* **9472**, p. 94720R, 2015.
18. M. Bertalmío, G. Sapiro, V. Caselles, and C. Ballester, “Image inpainting,” in *Proc. 27th annual conference on computer graphics and interactive techniques (SIGGRAPH)*, pp. 417–424, ACM Press/Addison-Wesley Publishing Co., (New York, NY, USA), 2000.

19. Z. Xu and J. Sun, "Image inpainting by patch propagation using patch sparsity," *IEEE Trans. Image Processing* **19**, pp. 1153–1165, 2010.
20. B. Wohlberg, "Inpainting by joint optimization of linear combinations of exemplars," *IEEE Signal Processing Letters* **18**, pp. 75–78, 2011.
21. J. Theiler and J. Bloch, "Multiple concentric annuli for characterizing spatially nonuniform backgrounds," *The Astrophysical Journal* **519**, pp. 372–388, 1999.
22. J. Theiler and J. Bloch, "Heuristic estimates of weighted binomial statistics for use in detecting rare point source transients," in *Astronomical Data Analysis Software and Systems VI, Astronomical Society of the Pacific Conference Series* **125**, pp. 151–154, 1997.
23. J. Theiler and J. Bloch, "Nested test for point sources," in *Statistical Challenges in Modern Astronomy II*, G. J. Babu and E. D. Feigelson, eds., pp. 407–408, Springer Verlag, (New York), 1997.
24. A. Schaum. Personal communication.
25. Airborne Visible/Infrared Imaging Spectrometer (AVIRIS), Jet Propulsion Laboratory (JPL), National Aeronautics and Space Administration (NASA) <http://aviris.jpl.nasa.gov/>.
26. G. Vane, R. O. Green, T. G. Chrien, H. T. Enmark, E. G. Hansen, and W. M. Porter, "The Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)," *Remote Sensing of the Environment* **44**, pp. 127–143, 1993.
27. D. Snyder, J. Kerekes, I. Fairweather, R. Crabtree, J. Shive, and S. Hager, "Development of a web-based application to evaluate target finding algorithms," *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* **2**, pp. 915–918, 2008.
28. "SpecTIR advanced hyperspectral and geospatial solutions." Free data samples [Online] <http://www.spectir.com/free-data-samples/>.
29. J. Theiler and D. Hush, "Statistics for characterizing data on the periphery," *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* , pp. 4764–4767, 2010.
30. J. Theiler, "Ellipsoid-simplex hybrid for hyperspectral anomaly detection," *Proc. 3rd IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* , 2011.
31. L. Bachega, J. Theiler, and C. A. Bouman, "Evaluating and improving local hyperspectral anomaly detectors," *Proc. 40th IEEE Applied Imagery and Pattern Recognition (AIPR) Workshop* , 2011.
32. G. Groszklos and J. Theiler, "Ellipsoids for anomaly detection in remote sensing imagery," *Proc. SPIE* **9472**, p. 94720P, 2015.
33. W. F. Basener, E. Nance, and J. Kerekes, "The target implant method for predicting target difficulty and detector performance in hyperspectral imagery," *Proc. SPIE* **8048**, p. 80481H, 2011.
34. J. Theiler, "Matched-pair machine learning," *Technometrics* **55**, pp. 536–547, 2013.
35. J. Theiler, "Transductive and matched-pair machine learning for difficult target detection problems," *Proc. SPIE* **9088**, p. 90880E, 2014.
36. J. Theiler, "By definition undefined: Adventures in anomaly (and anomalous change) detection," *Proc. 6th IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* , 2014.
37. S. Matteoli, M. Diani, and G. Corsini, "A tutorial overview of anomaly detection in hyperspectral images," *IEEE A&E Systems Magazine* **25**, pp. 5–27, 2010.
38. S. Matteoli, M. Diani, and J. Theiler, "An overview of background modeling for detection of targets and anomalies in hyperspectral remotely sensed imagery," *J. Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)* **7**, pp. 2317–2336, 2014.
39. P. Milanfar, "A tour of modern image filtering: New insights and methods, both practical and theoretical," *IEEE Signal Processing Magazine* **30**, pp. 106–128, Jan 2013.
40. B. Wohlberg, "Endogenous convolutional sparse representations for translation invariant image subspace models," *Proc. IEEE International Conference on Image Processing (ICIP)* , 2014.
41. B. Wohlberg, "Efficient convolutional sparse coding," *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* , pp. 7173–7177, 2014.