



Linear and kernel methods for multivariate change detection[☆]

Morton J. Canty^{a,*}, Allan A. Nielsen^b

^a Institute for Bio- and Geosciences, IBG 3, Jülich Research Center, D-52425 Jülich, Germany

^b National Space Institute, Technical University of Denmark, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, Denmark

ARTICLE INFO

Article history:

Received 6 December 2010

Received in revised form

15 March 2011

Accepted 15 May 2011

Available online 12 June 2011

Keywords:

CUDA

ENVI

IDL

IR-MAD

iMAD

Kernel methods

Matlab

Radiometric normalization

Remote sensing

Multiresolution

ABSTRACT

The iteratively reweighted multivariate alteration detection (IR-MAD) algorithm may be used both for unsupervised change detection in multi- and hyperspectral remote sensing imagery and for automatic radiometric normalization of multitemporal image sequences. Principal components analysis (PCA), as well as maximum autocorrelation factor (MAF) and minimum noise fraction (MNF) analyses of IR-MAD images, both linear and kernel-based (nonlinear), may further enhance change signals relative to no-change background. IDL (Interactive Data Language) implementations of IR-MAD, automatic radiometric normalization, and kernel PCA/MAF/MNF transformations are presented that function as transparent and fully integrated extensions of the ENVI remote sensing image analysis environment. The train/test approach to kernel PCA is evaluated against a Hebbian learning procedure. Matlab code is also available that allows fast data exploration and experimentation with smaller datasets. New, multiresolution versions of IR-MAD that accelerate convergence and that further reduce no-change background noise are introduced. Computationally expensive matrix diagonalization and kernel image projections are programmed to run on massively parallel CUDA-enabled graphics processors, when available, giving an order of magnitude enhancement in computational speed. The software is available from the authors' Web sites.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

In a standard change detection situation involving optical remote sensing imagery, two multi- or hyperspectral images of the same scene are acquired at two different points in time and then compared. Between acquisitions, ground reflectance changes will have occurred at some locations, but in general not everywhere. In order to observe the changes, the images are accurately registered to one another and—optionally—corrected for atmospheric and illumination effects. The necessary preprocessing steps having been performed, it is common to examine functions of the spectral bands (differences, ratios, or other linear or nonlinear band combinations) that bring change information contained within them to the fore. Singh (1989) gives a good, but now somewhat outdated, survey of change detection algorithms for remotely sensed data. For more recent reviews in a more general context, see Radke et al. (2005) or Coppin et al. (2004) and in the context of very-high-resolution imagery, Marchesi et al. (2010). Alternatively, the objective may not be to

observe change, but rather to eliminate relative differences between the images arising from effects due to the atmosphere, sensor gain, or differing solar illumination conditions. This can sometimes be achieved by linear radiometric normalization using invariant pixels identified within the images, that is, on the basis of no-change rather than change observations (Schott et al., 1988; Hall et al., 1991; Moran et al., 1992; Yang and Lo, 2000; Furby and Campbell, 2001; Du et al., 2002).

In a series of publications (Nielsen et al., 1998; Canty et al., 2004; Canty and Nielsen, 2006, 2008; Nielsen, 2007), the multivariate alteration detection (MAD) transformation and a modification involving iterative reweighting (IR-MAD or iMAD) were proposed, both for unsupervised change detection and for automatic radiometric normalization. More recently, Nielsen (2011) discussed, among other applications, the successful use of kernel versions of maximum autocorrelation factor (MAF) and minimum noise fraction (MNF) transformations for the postprocessing of difference images for change detection.

In this contribution, we present efficient and easy-to-use software implementations for IR-MAD and radiometric normalization, as well as for kernelized versions of principal components analysis (PCA), and the MAF and MNF transformations. The paper is organized as follows. In Section 2 we briefly outline the IR-MAD transformation, pointing out its advantages both for change detection and for radiometric normalization, and introduce new,

[☆] Code available from: <http://mcanty.homepage.t-online.de/software.html>, <http://www2.imm.dtu.dk/~aa/software.html>.

* Corresponding author. Tel.: +49 2461 51850; fax: +49 2461 612518.

E-mail addresses: m.canty@fz-juelich.de (M.J. Canty), aa@space.dtu.dk (A.A. Nielsen).

multiresolution variants of IR-MAD. In Section 3 the kernel methods are summarized. In Section 4 we outline some specific choices made in the software implementations and describe IDL and Matlab programs for IR-MAD, radiometric normalization, kernel PCA, MAF, and MNF. The IDL routines, which function as fully integrated extensions of the ENVI remote sensing image analysis environment, can run on conventional CPU architectures as well as take advantage of the massively parallel capabilities of graphics processors. In Section 5, examples illustrating IR-MAD applied to multispectral imagery and the postprocessing of change images with kernel transformations are presented and the adopted train/test approach to kernel transformations is examined. Conclusions are drawn in Section 6.

2. Change detection

The observations (pixel vectors) in a bitemporal, p -band, multispectral image may be represented by random vectors $\mathbf{X} = (X_1 \dots X_p)^T$ and $\mathbf{Y} = (Y_1 \dots Y_p)^T$ for the first and second acquisitions, respectively. The components X_i and Y_i correspond to the original spectral bands and are conventionally ordered by wavelength. The MAD algorithm determines transformation matrices

$$\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2 \dots \mathbf{a}_p), \quad \mathbf{B} = (\mathbf{b}_1, \mathbf{b}_2 \dots \mathbf{b}_p) \quad (1)$$

such that the components of the transformed random vectors $\mathbf{U} = \mathbf{A}^T \mathbf{X}$, $\mathbf{V} = \mathbf{B}^T \mathbf{Y}$ are ordered by similarity, where similarity is measured by positive band-wise linear correlation (Nielsen et al., 1998; Nielsen, 2007). The transformations are obtained by applying standard canonical correlation analysis (CCA) (Hotelling, 1936). The elements of the vectors \mathbf{U} and \mathbf{V} are referred to as the canonical variates.

Taking paired differences (in reverse order) of the canonical variates generates a sequence of transformed images

$$M_i = U_{p-i+1} - V_{p-i+1}, \quad i = 1, \dots, p, \quad (2)$$

referred to as the MAD variates. The MAD variates have statistical properties that make them very useful for visualizing and analyzing change information. Thus, for instance, they are uncorrelated, $\text{cov}(M_i, M_j) = 0$, $i \neq j$, and have variances given in terms of the canonical correlations ρ_i by

$$\text{var}(M_i) = \sigma_{M_i}^2 = 2(1 - \rho_{p-i+1}), \quad (3)$$

which, by virtue of the chosen ordering, are successively decreasing.

2.1. Iterative reweighting

If the scenes were acquired under similar illumination conditions and if no ground reflectance changes whatsoever occurred between the two acquisitions, then the only differences between them would be due to random effects such as instrument noise and atmospheric fluctuation. From the central limit theorem, we would expect that the histogram of any linear combination of spectral bands would be very nearly Gaussian. In particular, the MAD variates, being uncorrelated, should follow a multivariate normal distribution with diagonal variance–covariance matrix. Since MAD variates associated with genuine changes will deviate more or less strongly from such a distribution, we expect an improvement of the sensitivity of the MAD transformation if emphasis is placed on establishing an increasingly better background of no change against which to detect change. This can be done in an iteration scheme in which observations are weighted by the probability of no change, as determined in the preceding iteration, when the sample means and variance–covariance matrices for the next iteration are estimated, thus leading to the iteratively reweighted MAD (IR-MAD) algorithm (Nielsen, 2007).

The probability weights may be determined by observing that the sum of the squares of the standardized MAD variates represented by the random variable Z ,

$$Z = \sum_{i=1}^p \left(\frac{M_i}{\sigma_{M_i}} \right)^2, \quad (4)$$

where σ_{M_i} is given by Eq. (3), will be χ^2 distributed with p degrees of freedom in the absence of change (distribution function $P_{\chi^2, p}(z)$). Accordingly, each observation is weighted by a no-change probability given by

$$\Pr(\text{no change}) = 1 - P_{\chi^2, p}(z), \quad (5)$$

where z is the realization of the random variable Z . Other weighting schemes are possible, for instance using Gaussian mixture clustering of change/no-change observations (Canty and Nielsen, 2006). Iteration of the MAD transformation continues until some stopping criterion is met, such as lack of significant change in the canonical correlations.

2.2. Generalization

Convergence to a no-change background depends on the presence of a sufficiently large fraction of invariant pixels in the scene (Canty and Nielsen, 2008), so that application of IR-MAD to a bitemporal image in which the no-change background is very small may not give a satisfactory result. This can often be remedied by running IR-MAD on a manually chosen spatial subset for which the ratio of no-change to change is believed to be higher and then using the transformation coefficients obtained to generalize to the full scene. The minimum ratio of no-change to change required for successful conversion of IR-MAD is discussed in detail in Canty and Nielsen (2008).

2.3. Multiresolution

For large images, e.g., satellite full scenes, recalculation of probability weights in Eq. (5), and hence convergence of the algorithm, is slow. We have developed scaled, or multiresolution, variants of IR-MAD that accelerate convergence and at the same time reduce the noise in the no-change background pixels.

In the ENVI implementation, pyramid representations of the images are calculated to a given depth; that is, the spatial resolutions are degraded by factors of 1 (no degradation), 2, 4, etc. Starting at the lowest resolution, the IR-MAD algorithm is run to convergence and the MAD variates are then resampled to the next higher resolution. The IR-MAD algorithm is run again on the correspondingly higher resolution images in the pyramid, but allowing only those observations to participate that have change probabilities, as determined from the up-sampled MAD χ^2 values, that exceed some threshold (e.g., 0.9). This procedure is repeated until the original image resolution is reached. The effect of the scaling is to pass the “spatial awareness” achieved at coarser scales up to finer scales, but at the same time to allow the details of regions of significant change to be successively refined. Since only a fraction of the pixels are involved at each resolution, convergence is fast.

The Matlab implementation carries the weights (equal to the no-change probabilities) in IR-MAD across scale space from coarse to finer scales after letting the iterations run to convergence at each level in scale space. The coarse-scale versions can be calculated in several ways; here it is done by smoothing with a five-by-five Gaussian filter. Unlike the ENVI implementation, no subsampling is done to create a pyramid representation.

2.4. Radiometric normalization

The usefulness of the IR-MAD transformation for radiometric normalization stems from the fact that the MAD varies in Eq. (2) are invariant under linear or affine transformations of either or both of the original images (Nielsen et al., 1998; Canty et al., 2004). Given this linear invariance, we can select for radiometric normalization all pixels that satisfy $\text{Pr}(\text{no change}) < t$, where t is a decision threshold; see Eq. (5). The pixels so selected will correspond to invariant features as long as the overall radiometric differences between the two images can be attributed to linear effects. This means that the relative radiometric normalization procedure can be carried out fully automatically.

3. Kernel transformations for postprocessing

As opposed to linear spectral transformations (PCA, MAF/MNF), nonlinear transformations, especially kernel MAF/MNF analysis of difference images, have been found to give conspicuously better suppression of both noise and signal in the no-change background (Nielsen, 2011). The kernel versions of PCA and MAF/MNF handle nonlinearities by implicitly transforming data by nonlinear mapping functions ϕ into higher, even infinite, dimensional feature space and then performing a linear analysis in that space. We outline these transformations briefly in the following.

3.1. Kernel principal components analysis

The so-called primal form of linear PCA is the eigenvalue problem

$$\frac{1}{n-1} \mathcal{X}^T \mathcal{X} \mathbf{w} = \lambda \mathbf{w}, \quad (6)$$

where \mathcal{X} is an $n \times p$ data design matrix in which n p -component centered observation vectors \mathbf{x}_i are stored as rows. The variance-covariance matrix $\mathcal{X}^T \mathcal{X} / (n-1)$ is $p \times p$ symmetric positive definite. The dual form is obtained by multiplying Eq. (6) from the left by \mathcal{X} to give (λ now subsumes the factor $n-1$)

$$\mathcal{X} \mathcal{X}^T \mathbf{v} = \lambda \mathbf{v}, \quad (7)$$

where $\mathbf{v} \propto \mathcal{X} \mathbf{w}$. The so-called Gram matrix $\mathcal{X} \mathcal{X}^T$ is $n \times n$ symmetric positive semidefinite with (i, j) th element given by the inner product $\mathbf{x}_i^T \mathbf{x}_j$ of observations. The kernel formulation for PCA (Schölkopf et al., 1998) is obtained from the dual form by kernel substitution, replacing the inner products by kernel functions $\kappa(\mathbf{x}_i, \mathbf{x}_j)$. The kernel functions implicitly represent inner products $\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ of nonlinear mappings $\phi(\mathbf{x})$ of the observations \mathbf{x} to some higher dimensional feature space. Kernel PCA then consists of the solution of the symmetric eigenvalue problem

$$\mathcal{K} \mathbf{v} = \lambda \mathbf{v}, \quad (8)$$

where $(\mathcal{K})_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ and \mathcal{K} is assumed to correspond to column-centered (means-subtracted) observations $\phi(\mathbf{x})$ in the nonlinear feature space. The kernel matrix has n^2 elements, where n is the number of observations. Therefore it is necessary to subsample and train on only a small portion of observations in order to be able to carry out kernel PCA (and also MAF/MNF analysis) on the large numbers of pixel vectors involved in remote sensing imagery.

Alternatively, a kernel version of generalized Hebbian learning (Kim et al., 2005; Günter et al., 2007), called the kernel Hebbian algorithm (KHA), may be used. The KHA iteratively calculates the first $r < n$ kernel principal component projections on the basis of all of the data as

$$\mathbf{y}_j = \mathbf{A} \mathbf{\kappa}_j, \quad j = 1, \dots, r. \quad (9)$$

Here \mathbf{y}_j is a column vector consisting of the first r kernel principal components of the j th observation, $\mathbf{\kappa}_j$ is the j th column of the full kernel matrix, and \mathbf{A} is an $r \times n$ matrix of coefficients trained according to the update rule

$$\mathbf{A}_{i+1} = \mathbf{A}_i + \eta_i [\mathbf{y}_i \mathbf{e}_i^T - \text{LT}(\mathbf{y}_i^T \mathbf{y}_i) \mathbf{A}_i], \quad \mathbf{y}_i = \mathbf{A}_i (\mathcal{K})_{\cdot i}. \quad (10)$$

In this expression, \mathbf{A}_i signifies the coefficient matrix after presentation of the i th training observation, η_i is a (gradually decreasing) learning rate parameter, \mathbf{e}_i is a unit vector with a “1” at the i th position, and $\text{LT}(\cdot)$ returns the lower triangular portion of its matrix operand.

After a training phase, which may involve several passes through the entire set of n observations, Eq. (9) is used to project the image along the first r nonlinear principal directions. The kernel principal axes themselves, i.e., the eigenvectors \mathbf{w} in the nonlinear feature-space equivalent of Eq. (6), are not explicitly available. However, the first r eigenvalues are given by

$$\lambda_i = \sqrt{\frac{\mathbf{a}_i \mathcal{K} (\mathbf{a}_i \mathcal{K})^T}{\mathbf{a}_i \mathbf{a}_i^T}}, \quad i = 1, \dots, r, \quad (11)$$

where \mathbf{a}_i is the i th row of \mathbf{A} . These correspond to the solution of Eq. (8) with the full kernel matrix. We shall return to the KHA in Section 5 when we investigate the validity of subsampling for kernel projections of change images.

3.2. Kernel MAF/MNF

As in the case of kernel PCA, kernel versions of the maximum autocorrelation factor (MAF) analysis and minimum noise fraction (MNF) analysis based on the dual formulation and kernel substitution can be formulated (Nielsen, 2011). For the kernel MAF problem this results in the generalized eigenvalue problem

$$\mathcal{K}^2 \mathbf{v} = \lambda \mathcal{K}_\Delta \mathcal{K}_\Delta^T \mathbf{v}, \quad (12)$$

where \mathcal{K} is the kernel matrix defined in Section 3.1 and (non-symmetric) \mathcal{K}_Δ contains kernelized versions of differenced data. In Nielsen (2011), \mathcal{K}_Δ has elements

$$\kappa(\mathbf{x}_i(r), \mathbf{x}_j(r) - \mathbf{x}_j(r + \Delta)), \quad (13)$$

where r denotes position and Δ is a small spatial shift, corresponding to carrying out the differencing in original feature space followed by kernelization. In the current version of the software, \mathcal{K}_Δ may optionally be chosen to have elements

$$\kappa(\mathbf{x}_i(r), \mathbf{x}_j(r)) - \kappa(\mathbf{x}_i(r), \mathbf{x}_j(r + \Delta)) \quad (14)$$

corresponding to differencing in extended feature space, which is conceptually more satisfactory. Note that $\kappa(\mathbf{x}_i(r), \mathbf{x}_j(r))$ are the elements in \mathcal{K} . The autocorrelation that is maximized is $1 - 1/(2\lambda)$. Solution of the generalized symmetric eigenvalue problem, Eq. (12), is discussed in the Appendix.

Similarly, for kernel MNF analysis, we solve $\mathcal{K}^2 \mathbf{v} = \lambda \mathcal{K}_N \mathcal{K}_N^T \mathbf{v}$, where (nonsymmetric) \mathcal{K}_N contains kernelized versions of data, to estimate the noise part. The noise fraction that is minimized is $1/\lambda$.

Obviously, these transformations may be used for general feature generation, dimensionality reduction, etc., and not just for change detection postprocessing.

4. Software

High-level program scripts were written in IDL and Matlab to code the methods outlined in the two preceding sections and are described below in Sections 4.1 and 4.2, respectively. We present first some general design decisions made in the implementations.

Sign ambiguity: To avoid ambiguity in the signs of the canonical transformations and hence of the IR-MAD variates, it was required that the correlations of the canonical variates, like those of the original bitemporal image bands, be positive; i.e.,

$$\rho_i = \mathbf{a}_i^T \Sigma_{xy} \mathbf{b}_i > 0, \quad i = 1, \dots, p, \quad (15)$$

where Σ_{xy} is the covariance matrix for the images \mathbf{X} and \mathbf{Y} . This does not completely remove the ambiguity in the signs of the eigenvectors \mathbf{a}_i and \mathbf{b}_i , since if we reverse both simultaneously the condition is still met. The ambiguity was resolved in our programs by requiring that the sum of the correlations of the canonical variates \mathbf{U} with each of the components X_j of the first image, $j=1, \dots, p$, be likewise positive.

Regularization: To counter possible near-singularity problems in the solution of the generalized eigenvalue problems involved in CCA, particularly when the number of spectral bands p is large, the IR-MAD scripts allow regularization with length and minimum curvature regularization (Nielsen, 2007).

Orthogonal regression: Radiometric normalization (Section 2.4) is carried out by linear least-squares regression of the target image (the one being normalized) on the reference image. Ordinary least-squares regression allows for measurement uncertainty in one variable only, whereas here both variables have measurement uncertainty associated with them—in fact, which variable is termed reference and which is termed target data is arbitrary. Therefore orthogonal linear regression, which treats the data symmetrically, is applied for normalization (Canty et al., 2004).

Projecting and centering: For multispectral images we work with training and test data (this is typically needed for large data sets). For example, in the case of kernel PCA, having solved the eigenvalue problem (8) with training observations \mathbf{x}_j , $j=1, \dots, n$, we project all of the image pixels \mathbf{x}_v , $v=1, \dots, m$, along the first r principal directions in the nonlinear feature space according to

$$P_i[\phi(\mathbf{x}_v)] = \sum_{j=1}^n \frac{1}{\sqrt{\lambda_i}} (\mathbf{v}_i)_j \kappa(\mathbf{x}_j, \mathbf{x}_v), \quad i = 1, \dots, r, \quad v = 1, \dots, m, \quad (16)$$

where $(\lambda_i, \mathbf{v}_i)$ are eigenvalue/eigenvector pairs for (8) and $\kappa(\mathbf{x}_j, \mathbf{x}_v)$ is a centered kernel matrix. We may wish to center the test data with the training data mean

$$\bar{\phi}_{\text{train}} = \frac{1}{n} \sum_{i=1}^n \phi(\mathbf{x}_i)$$

or with the test data mean

$$\bar{\phi}_{\text{test}} = \frac{1}{m} \sum_{v=1}^m \phi(\mathbf{x}_v).$$

If the test data cannot be held in memory and we center the test data with the test data mean, we must kernelize the training data with the test data twice: once to calculate row and column means and once to actually center. If we center the test data with the training data mean, we need to kernelize the training data with the test data only once; the mean values needed come from the training kernel. Details are given in Nielsen (under review).

Parallelization: Kernel transformations are so-called memory-based methods, in the sense that the training data used to determine the transformation coefficients are also required for generalization. This implies in particular that the final projection of the image is computationally intensive. The projections, in turn, involve the multiplication of large matrices, an operation that can be carried out efficiently in a parallel computing architecture. The ENVI/IDL scripts for kernel PCA and kernel MAF have been written to take advantage of CUDA-enabled graphics processors (Halfhill, 2008), if present on the host computer. Use is made of the

high-level IDL bindings to CUDA provided by Tech-X corporation in their library GPULib.¹

4.1. ENVI/IDL

IDL (Interactive Data Language) is an array- and graphics-oriented programming language with a powerful interface (ENVI=Environment for Visualizing Images) for importing and analyzing remote sensing imagery. Since ENVI is itself written in IDL, it can be extended very easily to include new processing algorithms, and these can be integrated seamlessly into the ENVI menu system.

4.1.1. IR-MAD and radiometric normalization

Similarly to other linear spectral transformations included in the standard ENVI environment, such as PCA and MNF, there are two ways to run IR-MAD:

- (1) by computing image statistics from the current data, i.e., the bitemporal image itself, or
- (2) by reading an existing statistics file and applying it to the current data.

The first of these is the more common. After prompting for the (eventually masked) input bitemporal image, the IR-MAD program generates the canonical variates and the MAD variates together with a χ^2 image. The χ^2 image can be used for choosing invariant pixels for radiometric normalization on the basis of Eq. (5) as discussed below. A statistics file can also be generated, in which transformation coefficients calculated with the current data can be saved.

The second way to run IR-MAD is provided primarily for situations in which the algorithm fails to converge because the amount of real change between the acquisitions is too large, and a useful no-change background is not found; see Section 2.2. If this is the case, it may be possible, by experimentation, to find a spatial subset of the data for which convergence is satisfactory. The generated statistics file (see above) may then be used to generalize to the entire scene. After responding to the prompt for an existing statistics file, the user enters the (spatial/spectral) subsets of the two images (the spectral subsets must of course concur with those used to generate the statistics file). The MAD transformation is then performed immediately.

Image pyramids for multiresolution IR-MAD (Section 2.3) are generated by a partial discrete wavelet transform (PDWT) with Daubechies wavelets (Daubechies, 1988; Mallat, 1989). The PDWT uses a recursive filter bank and does not require additional storage. The inverted filter bank losslessly reconstructs progressively higher resolution images as required by the algorithm. The multiresolution IR-MAD algorithm is also callable directly from the ENVI menu and, apart from prompting for the desired pyramid depth, follows essentially the same input/output conventions.

The input images may optionally be masked. There are two primary reasons for masking the images: (i) “Black edge” pixels. Often, when full scenes are processed, the image margins contain no data. The MAD algorithm may then misinterpret these pixels as no-change background and converge to them. Since the edge pixels are constants (usually zeroes), the weighted covariance matrix will quickly become degenerate and the program will abort. (ii) Large water bodies. Generally, water bodies provide a good no-change background against which to measure change. However, if they are large and if illumination effects (e.g., due to waves, solar glare) lead to a uniform difference in reflectance

¹ <http://www.txcorp.com/products/GPULib/>.

between the two acquisitions, then they too can constitute a false no-change background to which the MAD algorithm may converge. Both effects can be countered by appropriate masking. Cloud cover, on the other hand, is usually not a problem, since it corresponds to genuine change.

The IDL code takes advantage of ENVI's built-in functionality to process images of virtually any size. The second-order statistics (means and covariance matrices) needed for CCA are calculated by sampling all of the pixels in both input images. To this end, the image pixels are read in row by row using ENVI's "spectral tiling" facility, and the statistics are updated with the method of provisional means. Since the latter algorithm is iterative and would require an inefficient IDL FOR-loop over the pixels in each row, it is programmed in C as a so-called dynamic load module (DLM) extension to IDL.

The radiometric normalization program can be invoked for the same image pair after IR-MAD has been run. The user is again prompted to select (a spatial/spectral subset of) the two multispectral images, first the reference image and then the one to be normalized. Next he or she chooses the corresponding χ^2 image generated previously by IR-MAD and the minimum probability to use to identify no-change pixels. After completion, another image (e.g., a full scene) may be normalized to the reference with the regression coefficients just determined. This is convenient, for example, if two images with only a partial overlap are to be mosaicked.

4.1.2. Kernel transformations

The ENVI/IDL extensions for kernel PCA and kernel MAF may also be called from the ENVI main menu. The user is queried for an input file, a training sample size, the number of transformed components to retain, the kernel type, and the associated kernel parameters. The available kernels are

$$k_{\text{lin}}(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j,$$

$$k_{\text{poly}}(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + r)^d,$$

$$k_{\text{rbf}}(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2),$$

$$k_{\text{sig}}(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i^T \mathbf{x}_j + r).$$

Choosing the linear kernel is, apart from the effect of subsampling, equivalent to running linear PCA or linear MAF. The Gaussian kernel (k_{rbf} above) is the default. For that kernel, the parameter γ essentially determines the training/generalization tradeoff, with large values leading to overfitting (Shawe-Taylor and Cristianini, 2004). It is calculated in terms of a user-defined parameter N_{SCALE} as

$$\gamma = \frac{1}{2(N_{\text{SCALE}} \sigma)^2}, \quad (17)$$

where $\sigma = \langle \|\mathbf{x}_i - \mathbf{x}_j\| \rangle_{i \neq j}$ is the average Euclidean distance between the training observations.

The Gaussian kernel matrix can be calculated efficiently in an array-oriented language such as IDL. If a CUDA-enabled graphics device (GPU) is present, kernel matrix evaluation in IDL can be speeded up considerably with the help of the IDL bindings made available in the GPULib library. The data matrices are transferred to the graphics device with the aid of GPULib procedures. GPULib routines then work exclusively with device pointers (handles) so that all computations are performed on the GPU in code optimized for parallel processing. A handle to the kernel matrix, still residing in graphics memory, is then returned. Avoiding the bandwidth limitations of host \leftrightarrow device transfers is an important design consideration in GPULib. A large palette of GPU counterparts of standard IDL functions has been provided in order to

allow as much processing as possible to take place on the graphics device before results are returned to the CPU.

After centering of the kernel and solution of the appropriate eigenvalue problems, the projection is carried out in one or two passes through the image. If the test data are centered on the test mean, then, on a first pass, the matrix column, row, and overall sums required for centering are accumulated. These are applied on the second pass as each image pixel is projected. For centering on the training mean, the first pass is unnecessary, as discussed earlier. If CUDA is available, then both centering and projection are performed entirely on the graphics device, resulting in an order-of-magnitude reduction in processing time. (These operations can be carried out in single precision.) Otherwise the host CPU is used. The kernel transformation programs do not make use of the ENVI tiling facility, as they are not intended to be used with very large images.

For rank determination in the kernel MAF transformation, calculation of all of the eigenvalues of $\mathbf{K}_A \mathbf{K}_A^T$ is required; see Eq. (12). This can also be relegated to the GPU by making use of the CULA Tools library,² which ports LAPACK routines to CUDA. In this case, the graphics processor must be capable of double precision operations.

4.2. Matlab

The Matlab code provided holds everything in memory and is meant for experimentation on smaller images, not for production runs on full scenes. It does not make use of graphics hardware for parallel acceleration of the computations. Otherwise it provides the same functionality as the ENVI/IDL code and is very easily changed to try out new ideas. For reasons of space it will not be described further.

5. Examples

In previous publications, several studies of the application of IR-MAD and its associated automatic radiometric normalization to multi- and hypervariate imagery have been given (Canty et al., 2004; Canty and Nielsen, 2006, 2008; Nielsen, 2007). Therefore, in this section, we restrict ourselves to examples involving the new multiresolution algorithms (Section 2.3) and kernel postprocessing (Section 3).

5.1. Multiresolution IR-MAD

The multiresolution algorithm implemented in ENVI/IDL is compared with standard IR-MAD using the Landsat 5 TM bitemporal scene shown in Fig. 1. The two images were acquired within about seven weeks of each other, with changes occurring in the extent of a reservoir (shallow flooding) and in agricultural areas to the north. Further changes in the reservoir are likely caused by phenological effects or higher sediments in the water after a heavy rain or vegetation growth.

The χ^2 image of the IR-MAD variates (see Eq. (4)) is shown in Fig. 2, where the scaling is seen to reduce the noise in the no-change background (black areas). Table 1 compares the signal-to-noise ratios in all six MAD bands. Noise statistics were estimated on the basis of differences of one-pixel shifts. A change probability threshold of 0.9 was used to decide which observations participate in the successive refinements. The results were found to be fairly insensitive to the threshold chosen, with similar noise reductions obtaining for values between 0.85 and 0.95.

² <http://www.culatools.com/>.

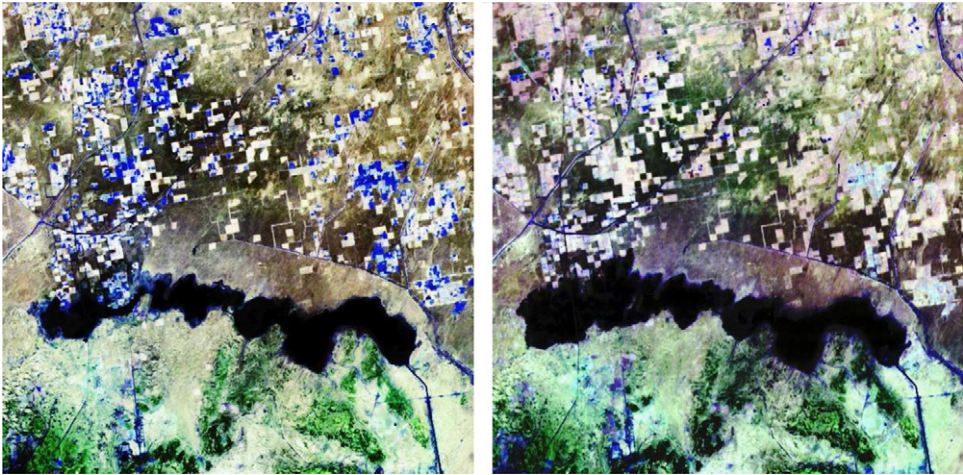


Fig. 1. Bitemporal scene over a water reservoir in India. Landsat 5 Thematic Mapper acquired on 29 March 1998 (left) and 16 May 1998 (right). The images are displayed as RGB composites of bands 7, 5, and 4 in a histogram equalization stretch.

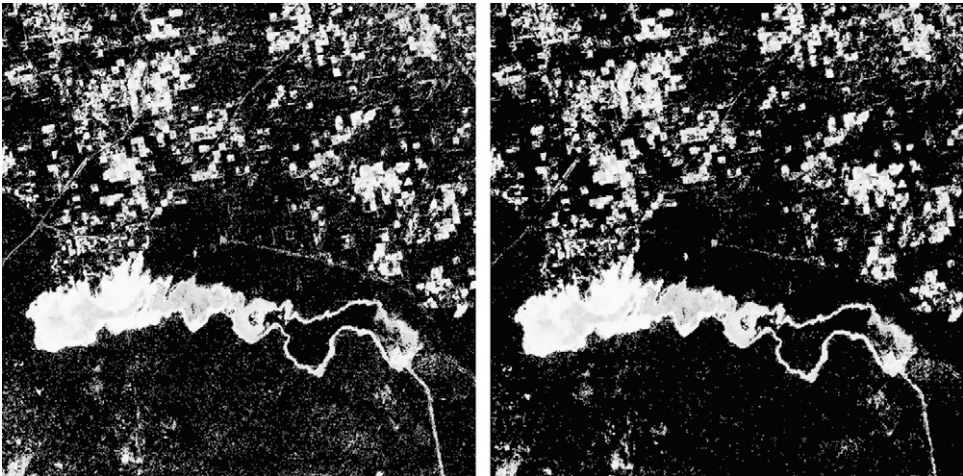


Fig. 2. χ^2 images for IR-MAD transformations of a bitemporal scene of Fig. 3. Left: standard IR-MAD. Right: multiresolution algorithm with pyramid depth 2.

Table 1
Signal-to-noise ratios for the IR-MAD variates for the bitemporal image of Fig. 3.

Algorithm	MAD1	MAD2	MAD3	MAD4	MAD5	MAD6
Multires.	2.3	3.8	11.1	13.4	7.9	42.9
Standard	1.1	2.2	8.4	7.8	4.8	34.5

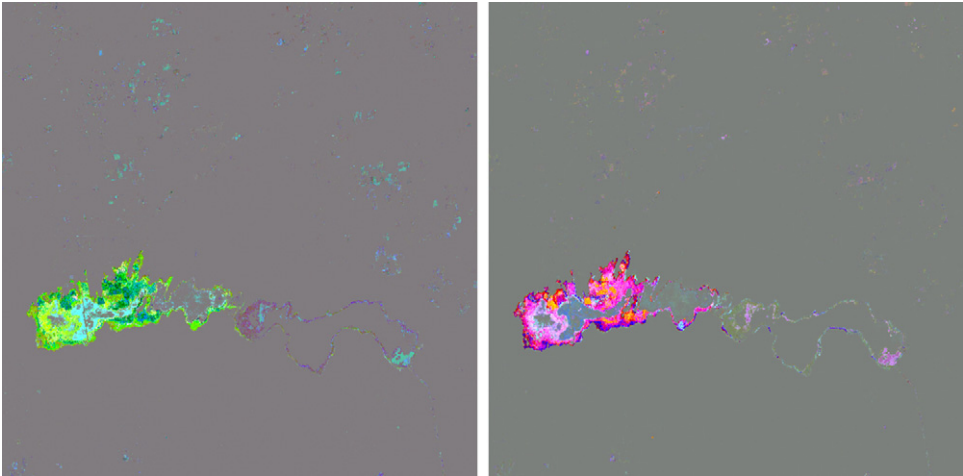


Fig. 3. Kernel MAF variates 1, 2, and 3 of all IR-MAD variates as RGB (left) for the bitemporal scene of Fig. 1; kernel MNF variates 1, 2, and 3 of all IR-MAD variates as RGB (right).

5.2. Kernel MAF/MNF

As an example of kernel MAF and MNF postprocessing of change images, we use the same imagery as in Fig. 2. Fig. 3 left shows an RGB representation of kernel MAF variates 1, 2, and 3 of all six IR-MAD variates from a standard analysis; i.e., the multi-resolution version is not applied here. Fig. 3 right shows an RGB representation of kernel MNF variates 1, 2, and 3 of the same six IR-MAD variates. All variates are stretched linearly between mean value minus and mean value plus six standard deviations. Approximately 1000 training samples are used to calculate the transforms applied. The Gaussian kernel was used with a parameter γ as determined by (17) with $NSCALE=1$, so that its σ is the average distance between training observations in the original feature space. This is a typical value that ensures that the nonlinearity of the Gaussian kernel is effective.

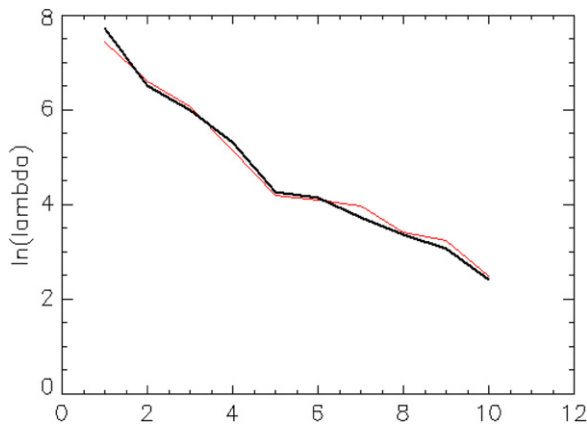


Fig. 4. The first 10 eigenvalues of the kernel PCA for a (100×100) -pixel, six-band Landsat 7 ETM+ image. Black curve: with 1% subsampling. Red curve: with KHA after 50 passes through the dataset. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

As is the case for the IR-MAD variates, in these images areas with saturated colors (including black and white, if present) are change regions; grayish regions are no-change. Note how both kernel MAF analysis and kernel MNF analysis focus on the extreme change observations. Also, although the coloring of change pixels is different, it is the same pixels that are highlighted as representing change.

The effect of subsampling for kernel spectral transformations can be examined, in the case of kernel PCA, by comparison with the KHA method (Section 3.1), which generates transformations on the basis of all of the pixel data. Fig. 4 compares the largest eigenvalues for kernel PCA applied to a small Landsat 7 ETM+ image using a 1% subsample followed by diagonalization of the kernel matrix with those obtained from KHA. Fig. 5 compares the eigenvectors (projection directions in nonlinear feature space) on the basis of scatterplots of principal component projections for subsampling and KHA. Correlations begin to deteriorate at the fifth or sixth eigenvector.

6. Conclusion

We have presented and illustrated efficient and easy-to-use IDL and Matlab software for multivariate change detection and radiometric normalization as well as for kernelized versions of principal components, maximum autocorrelation factors, and maximum noise fraction transformations. Comparison with the kernel Hebbian algorithm indicates that the use of 1% subsampling for kernel methods will give satisfactorily reproducible results for the first five or six eigenvectors. We have also introduced new, multiresolution variants of the IR-MAD algorithm, together with IDL and Matlab code. The IDL programs will take advantage of the parallel processing capabilities of CUDA-enabled graphics processors when they are available. The software may be obtained from the authors' Web sites:

IDL/ENVI: <http://mcanty.homepage.t-online.de/software.html>

Matlab: <http://www2.imm.dtu.dk/~aa/software.html>.

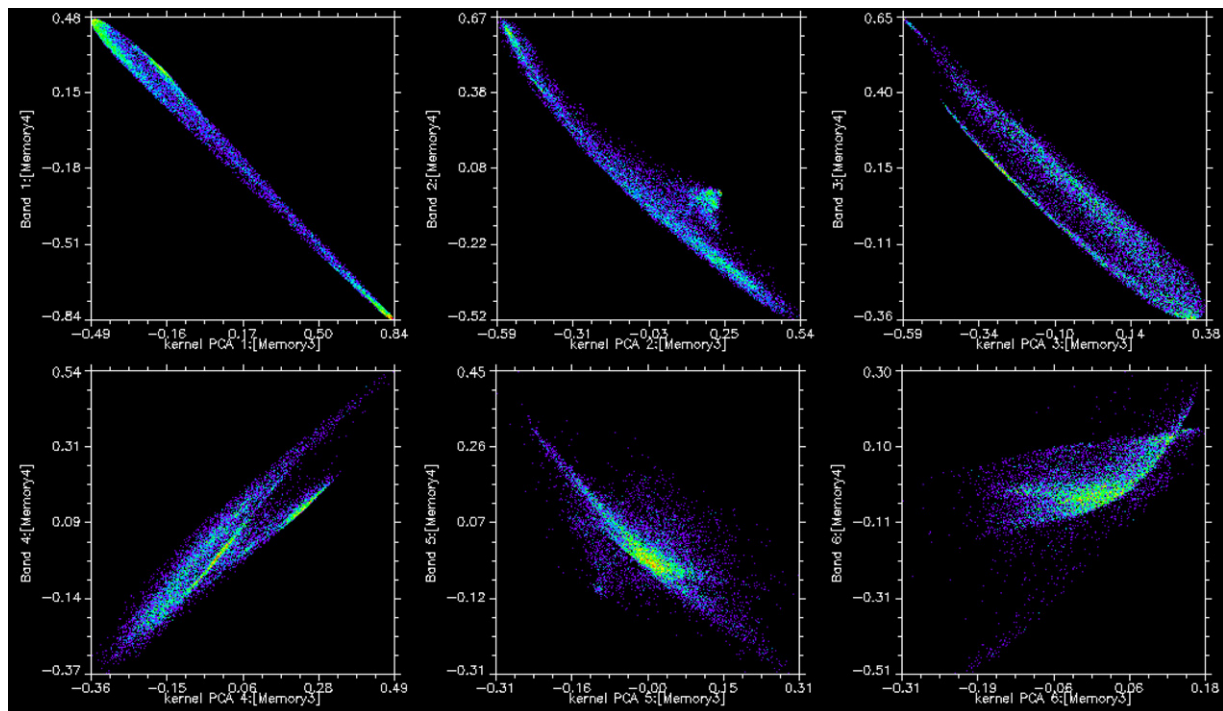


Fig. 5. Scatterplots of the first six kernel principal components calculated with 1% subsampling and kernel matrix diagonalization (x-axis) with those calculated with KHA after 50 passes through the dataset (y-axis); see Fig. 4.

Users of the programs should acknowledge the source by citing the relevant publications.

Acknowledgment

Thanks to Dr. Luis Gomez-Chova, University of Valencia, Spain, for suggesting the centering of the test data with the training data mean.

Appendix A

The symmetric generalized eigenvalue problem may be solved by writing the symmetric right hand side matrix as a product of matrix square roots,

$$\mathbf{A}\mathbf{w} = \lambda\mathbf{B}\mathbf{w} = \lambda\mathbf{B}^{1/2}\mathbf{B}^{1/2}\mathbf{w},$$

where $\mathbf{B}^{1/2} = \mathbf{P}\mathbf{\Lambda}^{1/2}\mathbf{P}^T$, with \mathbf{P} consisting of columns of eigenvectors, and $\mathbf{\Lambda}^{1/2}$ is a diagonal matrix of square roots of the eigenvalues of \mathbf{B} . If \mathbf{B} is full rank, $r=n$, we retain all columns and all rows of both \mathbf{P} and $\mathbf{\Lambda}$. If \mathbf{B} has rank $r < n$ we retain only the first r columns corresponding to the highest eigenvalues (but all rows) of \mathbf{P} and only the r first rows and r first columns of $\mathbf{\Lambda}$. Since $\mathbf{P}^T\mathbf{P} = \mathbf{I}_r$ (and $\mathbf{P}\mathbf{P}^T = \mathbf{I}_n$), this leads to the desired $\mathbf{B} = \mathbf{P}\mathbf{\Lambda}^{1/2}\mathbf{P}^T$ $\mathbf{P}\mathbf{\Lambda}^{1/2}\mathbf{P}^T = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^T$. The problem now rewrites to

$$(\mathbf{B}^{-1/2}\mathbf{A}\mathbf{B}^{-1/2})(\mathbf{B}^{1/2}\mathbf{w}) = \lambda(\mathbf{B}^{1/2}\mathbf{w}),$$

which is a symmetric ordinary eigenvalue problem. In this case we may get the inverse for $\mathbf{B}^{1/2}$ as $\mathbf{B}^{-1/2} = (\mathbf{P}\mathbf{\Lambda}^{1/2}\mathbf{P}^T)^{-1} = \mathbf{P}\mathbf{\Lambda}^{-1/2}\mathbf{P}^T$, where $\mathbf{\Lambda}^{-1/2}$ is an r by r diagonal matrix of inverse square roots of the eigenvalues.

The IDL and Matlab code solves the above problem, normalizes the eigenvectors so that the kernel MAF variates have unit variance, and calculates the kernel MAFs.

Appendix B. Supplementary data

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.cageo.2011.05.012.

References

- Canty, M.J., Nielsen, A.A., 2006. Visualization and unsupervised classification of changes in multispectral satellite imagery. *International Journal of Remote Sensing* 27 (18), 3961–3975. Available at <http://www.imm.dtu.dk/pubdb/p.php?3389>.
- Canty, M.J., Nielsen, A.A., 2008. Automatic radiometric normalization of multi-temporal satellite imagery with the iteratively re-weighted MAD transformation. *Remote Sensing of Environment* 112 (3), 1025–1036. Available at <http://www.imm.dtu.dk/pubdb/p.php?5362>.
- Canty, M.J., Nielsen, A.A., Schmidt, M., 2004. Automatic radiometric normalization of multitemporal satellite imagery. *Remote Sensing of Environment* 91 (3–4), 441–451. Available at <http://www.imm.dtu.dk/pubdb/p.php?2815>.
- Coppin, P., Jonckheere, I., Nackaerts, K., Muys, B., 2004. Digital change detection methods in ecosystem monitoring: a review. *International Journal of Remote Sensing* 25 (9), 1565–1596.
- Daubechies, I., 1988. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics* 41, 909–996.
- Du, Y., Teillet, P.M., Cihlar, J., 2002. Radiometric normalization of multitemporal high-resolution images with quality control for land cover change detection. *Remote Sensing of Environment* 82, 123–134.
- Furby, S.L., Campbell, N.A., 2001. Calibrating images from different dates to like-value counts. *Remote Sensing of Environment* 77, 186–196.
- Günter, S., Schraudolph, N.N., Vishwanathan, S.V.N., 2007. Fast iterative kernel principal component analysis. *Journal of Machine Learning Research* 8, 1893–1918.
- Halfhill, T.R., 2008. Parallel processing with CUDA. In: *Microprocessor Report*, Reed Electronics, Scottsdale, Az, pp. 1–8.
- Hall, F.G., Strebel, D.E., Nickeson, J.E., Goetz, S.J., 1991. Radiometric rectification: Toward a common radiometric response among multirate, multisensor images. *Remote Sensing of Environment* 35, 11–27.
- Hotelling, H., 1936. Relations between two sets of variates. *Biometrika* 28, 321–377.
- Kim, K.I., Franz, M.O., Schölkopf, B., 2005. Iterative kernel principal component analysis for image modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (9), 1351–1366.
- Mallat, S.G., 1989. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11 (7), 674–693.
- Marchesi, S., Bovolo, F., Bruzzone, L., 2010. A context-sensitive technique robust to registration noise for change detection in VHR multispectral images. *IEEE Transactions on Image Processing* 19 (7), 1877–1889.
- Moran, M.S., Jackson, R.D., Slater, P.N., Teillet, P.M., 1992. Evaluation of simplified procedures for retrieval of land surface reflectance factors from satellite sensor output. *Remote Sensing of Environment* 41, 160–184.
- Nielsen, A.A., 2007. The regularized iteratively reweighted MAD method for change detection in multi- and hyperspectral data. *IEEE Transactions on Image Processing* 16 (2), 463–478. Available at <http://www.imm.dtu.dk/pubdb/p.php?4695>.
- Nielsen, A.A., 2011. Kernel maximum autocorrelation factor and minimum noise fraction transformations. *IEEE Transactions on Image Processing* 20 (3), 612–624. Available at <http://www.imm.dtu.dk/pubdb/p.php?5925>.
- Nielsen, A.A. The kernel MAF and MNF transformations revisited. *IEEE Transactions on Signal Processing*, under review.
- Nielsen, A.A., Conradsen, K., Simpson, J.J., 1998. Multivariate alteration detection (MAD) and MAF post-processing in multispectral, bitemporal image data: New approaches to change detection studies. *Remote Sensing of Environment* 64, 1–19. Available at <http://www.imm.dtu.dk/pubdb/p.php?1220>.
- Radke, R.J., Andra, S., Al-Kofahi, O., Roysam, B., 2005. Image change detection algorithms: A systematic survey. *IEEE Transactions on Image Processing* 14 (4), 294–307.
- Schölkopf, B., Smola, A., Müller, K.-R., 1998. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation* 10 (5), 1299–1319.
- Schott, J.R., Salvaggio, C., Volchok, W.J., 1988. Radiometric scene normalization using pseudo-invariant features. *Remote Sensing of Environment* 26, 1–16.
- Shawe-Taylor, J., Cristianini, N., 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge, UK.
- Singh, A., 1989. Digital change detection techniques using remotely-sensed data. *International Journal of Remote Sensing* 10 (6), 989–1002.
- Yang, X., Lo, C.P., 2000. Relative radiometric normalization performance for change detection from multi-date satellite images. *Photogrammetric Engineering and Remote Sensing* 66, 967–980.