Initial Study of Schroedinger Eigenmaps for Spectral Target Detection

Leidy P. Dorado-Munoz[a], **David W. Messinger**[a]
[a]Rochester Institute of Technology, Center for Imaging Science, Digital Imaging and Remote Sensing Lab, Rochester, NY 14623 USA

Abstract. Spectral target detection refers to the process of searching for a specific material with a known spectrum over a large area containing materials with different spectral signatures. Traditional target detection methods in hyperspectral imagery (HSI) require assuming the data fits some statistical or geometric models and based on the model, to estimate parameters for defining a hypothesis test, where one class (i.e. target class) is chosen over the other classes (i.e. background class). Non-linear manifold learning methods such as Laplacian eigenmaps (LE) have extensively shown their potential use in HSI processing, specifically in classification or segmentation. Recently, Schroedinger eigenmaps (SE), which is built upon LE, has been introduced as a semi-supervised classification method. In SE, the former Laplacian operator is replaced by the Schroedinger operator. The Schroedinger operator includes by definition, a potential term $\mathbf{V}$ that steers the transformation in certain directions improving the separability between classes. In this regard, we propose a methodology for target detection that is not based on the traditional schemes and that does not need the estimation of statistical or geometric parameters. This novel method is based on SE, where the potential term, $\mathbf{V}$, is taken into consideration to include the prior knowledge about the target class and use it to steer the transformation in directions where the target location in the new space is known and the separability between target and background is augmented. An initial study of how SE can be used in a target detection scheme for HSI is shown here. In-scene pixel and spectral signature detection approaches are presented. The HSI data used comprises various target panels for testing simultaneous detection of multiple objects with different complexity.

Keywords: Schroedinger eigenmaps; Laplacian eigenmaps; Spectral graph theory; Spectral target detection; Hyperspectral image analysis.

*Leidy Dorado-Munoz, lpd5259@rit.edu

1 Introduction

Target detection is one of the most widely used applications of spectral remote sensing. It has a special importance due to its applicability in military surveillance and global environmental monitoring. Target detection methodologies address the problem of searching for a specific material with a known spectrum immersed in a group of objects or materials with different spectral signatures. The spectral information provided by hyperspectral imagery (HSI) is taken into consideration to get a better discrimination between pixels containing the material of interest and pixels that do not contain it. The traditional target detectors assume statistical or geometric models and estimate parameters based on those models, which are used to define a hypothesis test, where the material of interest is chosen over the rest of materials grouped into one class, i.e. the background class.[1]

The inherent variability of HSI, because of irregular material surfaces, spatial resolution vs. size of target objects, etc., introduces some difficulties that make the detection problem more complex. In some cases, the data models assumed by the traditional detection algorithms do not appropriately fit the structure of the data, resulting in poor detection performance. From another point of view, the detection problem can be seen as a binary classification, where it is possible to have the advantage of using novel mathematical models that have shown their potential use in HSI clustering or classification. These novel models are based on non-linear manifold learning algorithms, where the data is assumed to lie on or near to a lower dimensional manifold that encloses as best as possible, the local geometric structure of the data.[2] The linear manifold learning algorithms, such as Principal Component Analysis (PCA), assume that the data is linearly represented in the lower dimensional space describing as much of the data variability as possible.[3] This linearity leads to errors in the discrimination of materials, especially when the data is complex. This is the typical case of HSI, where the data collection system provides highly dense data with redundant information, which hides relevant features and whose analysis involves long processing time and high computational cost.[4] In general, non-linear manifold algorithms have been useful in cases where complex data is dealt with[5] and have shown to have a better performance in the clustering and discrimination of different classes in HSI, in spite of the processing costs that they exhibit. These algorithms attempt to preserve characteristics of the data, such as the local structure while the transformation is performed.

Many of these non-linear transformations techniques involve as a first step, the construction of a graph where each pixel is connected to a certain number of neighbors, and the data transformation ensures that the inter-pixel distances are preserved as best as possible in the new dimensional space. One of these non-linear transformations is *Laplacian eigenmaps* (LE), which assumes that the original data lies on a low dimensional manifold embedded in a high dimensional space. The LE method involves the calculation of the Laplace-Beltrami operator based on information recollected from the initial graph, and the representation of the data in the new space after the transformation is given by the eigenfunctions or eigenmaps of the Laplace operator. These eigenmaps are an approximation to a natural map defined on an entire manifold.[6] In this way, LE has been used in unsupervised classification problems due to its inherent preservation of the local structure leading to a natural clustering of the data. A semi-supervised classification scheme based on LE is *Schroedinger eigenmaps* (SE), which is built upon LE. SE, similarly to LE, involves the calculation of an operator, the Schroedinger operator,[7,8] that in addition to the connectivity information contained in the Laplacian operator also includes a potential term that steers the transformation in certain directions according to the *a priori* knowledge about the classes with the idea of increasing the separability among them.

Given this property of SE and considering the target problem as a binary classification between the target and background classes, the use of SE in target detection is proposed here. The idea is to include the prior knowledge about the target class in the potential term and steer the transformation in some directions where the target is separated from the background and, based on the properties of the transformation, is pulled toward the origin in the Schroedinger space. This fact could be taken into consideration to define a detection operator that scores high values to the points near zero and scores low values to those points far from zero. This definition of a detector that does not need to make a statistical or geometric assumption about the data presupposes differences with the typical detectors. Figure 1 shows the general overview of the traditional detection methodology

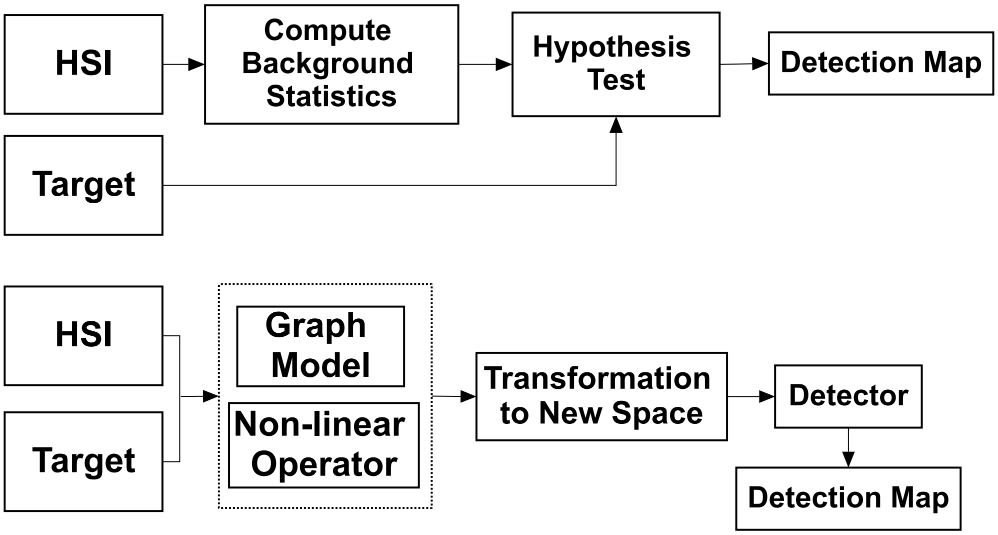


Fig 1 Traditional target detection methodology (top) vs. the proposed target detection methodology based on SE (bottom).

and the one based on SE. The figure presents the differences between the two schemes, emphasizing the novelties of the proposed method. The traditional approach defines a binary hypothesis, which requires estimating parameters based on statistical or geometric models. In contrast, the approach based on SE does not need to assume neither typical data models nor statistical/geometric parameters.

A novel target detection methodology based on SE is presented in this paper. Particularly, the use of a barrier potential as constraints that guides target like pixels toward a specific region in the transformed space is proposed. In addition, a study about the impact of the graph construction in SE and how this influences the separation of the target class is conducted. Results for in-scene target pixel detection and target signature detection are presented. For the assessment, a HSI scene from the RIT SHARE 2012 data acquisition campaign is used. The assessment includes a visual analysis, the use of False Alarm rate (FAr) as the metric to quantitatively assess the different approaches based on SE, and a comparison of the proposed methodology with the Adaptive Coherence Estimator (ACE) detector.[9]

1.1 Related Work

As was established in the introduction, target detection attempts to find a material with a specific and known spectrum in an image that has many materials with different spectral signatures. Due to the nature of the target detection problem and HSI data, the detection process faces problems related to the small number of pixels that could contain the target class compared to the total number of pixels in the image. In addition, searching for a specific material over a large area establishes practical difficulties which has driven the research community to propose different target detection techniques using HSI.[1] The diverse methods could be grouped into two categories, *unstructured* and *structured*, depending on the type of model they assume about the data set and specifically about the background. Structured methods assume geometric models such that the data set is a linear combination of basis-vector such as linear mixing models or linear subspace models. The main detector based on these geometric models is the Orthogonal Subspace Projector (OSP).[10] OSP inhibits as much as possible the background interference and enhances the target contribution by means of a data projection onto the subspace orthogonal to the background, where the residual energy of the target is high. However, if the number of background basis vectors is overestimated, it is possible that the background model includes target-signal components, which contributes to reduced target residual energy in the orthogonal space. Meanwhile, unstructured methods are based on statistical assumptions and the background is modeled as a single multivariate Gaussian random distribution.[1,11] Statistical parameters such as mean signature and covariance of the background are globally estimated from the entire image, and the same background covariance is assigned to the target. One of these detectors based on statistical estimations is the Adaptive Coherence Estimator (ACE),[12] which is a variation of the generalized likelihood ratio (GLR) and whose response is invariant to relative scaling of input data. There are also some variations introduced to these methodologies based on statistical assumptions. For instance in Ref.,[13] local estimations are performed by spectrally clustering the image and assuming a Gaussian distribution in each cluster.

On the other hand, non-linear manifold transformation methods have widely shown potential use in HSI processing, especially in classification or segmentation. Among non-linear manifold techniques used in clustering problems are diffusion maps,[14] isometric mapping,[15] local linear embedding,[16] and Laplacian eigenmaps.[6] All these techniques as non-linear transformations assume that the original data lies on a low dimensional manifold embedded in a high dimensional space, making it possible to find a meaningful representation of the original data in a lower dimensional space, where the local structure is preserved. Particularly, LE was first introduced by Belkin et.al. (2001)[6] as a method for embedding high dimensional data in a lower dimension. The core of LE is the Laplacian matrix which is an approximation of the Laplace-Beltrami operator that is defined on a manifold. A theoretical analysis to prove that eigenvectors of the graph Laplacian matrix converge to eigenfunctions of the Laplace-Beltrami operator on the underlying manifold is shown in Belkin and Niyogi (2006).[17] These eigenmaps are then the representation of the original data in the Laplacian space. This Laplacian representation, due to the tendency of keeping the local structure, leads to a natural clustering of the data which is taken into advantage in unsupervised classification,[6] in semi-supervised classification,[18,19] and segmentation problems.[20,21] The most used algorithm based on LE is *normalized cuts*, which addresses the segmentation problem as a graph partitioning problem by minimizing the normalized cut, a measure of disassociation between subgroups of a graph.[22] Many other algorithms based on LE have introduced different ways to build the graph in order to improve the segmentation or classification performance. For example, in Gillis and Bowles (2012),[20] spatial and spectral information is used in the construction of the graph; in Yang (2005),[23] different distance metrics are used to find the k nearest neighbors and build the graph. Also, some adaptive algorithms for searching for the number of nearest neighbor have been introduced in Zou et al. (2011)[24] and Mercovich et al. (2011),[25] where the general idea is to find an optimal number of nearest neighbors for each data point.

Other research projects have been more focused on the use of labeled and unlabeled data in order to improve the classification accuracy.[26] Regarding this, SE has been introduced as a scheme for semi-supervised classification.[7] As previously established, the Schroedinger matrix is built upon the Laplacian matrix. By definition, the Schroedinger matrix includes a potential term which could be used to encode the information about the position of labeled data. This potential term "steers" the diffusion process in the Schroedinger space to certain directions that produces the separation of data points that should not be in the same class, so correct clusters containing the labeled data are identified.[7] This SE methodology has also been proposed for use in remote sensing imagery,[8] and two kinds of potential terms have been proposed, the barrier potential and cluster potential, to identify different classes and achieve a good separability between them. SE has also been proposed for clustering by using spatial-spectral information, where the Laplacian operator includes the spectral similarity of the data pixels and the potential term in turn includes the spatial similarity.[27]

Although these non-linear techniques have been widely applied to classification, their use in target detection has been limited as a pre-processing step. In these methods, the data in the spectral space is transformed to a new space with the purpose of exposing hidden characteristics that may improve the detection. Once the data is represented in the new space, the typical detectors based on geometric or statistical models are used as detection operators for identifying the target class. In general, these techniques based on non-linear transformations use a graph model to represent the data structure and manifold learning methods to transform the data to a space favoring the identification of the target. For instance, Albano et al.[28] proposed the use of the commute time distance transformation as a non-linear transformation that separates outliers from the background. Ziemman and Messinger[29] proposed the use of Locally Linear Embedding (LLE) as the transformation algorithm, which is applied to both the image and the target signature. Contrary to those approaches, the use of SE in target detection offers not only the advantage of influencing the transformation in certain directions –favoring the separability of the target class from the background– but also to know where in the new space (i.e. Schroedinger space) the target class will be. Below, we take advantage of this fact to define a detection operator without assuming any statistical or geometric models.

1.2 Overview

Section 2.1 introduces the basic theory behind Schroedinger eigenmaps, including some basic terminologies related to graph theory and Laplacian eigenmaps. Section 3 presents the methodology and define the detector based on Schroedinger eigenmaps for spectral target detection. Section 4 shows the SEs performance in target detection and compares it with the ACE detector, which is widely used in target detection. Section 5 summarizes the results shown in Sec. 4, concludes the findings of this work and presents the future work.

2 Spectral Graph Theory: Laplacian and Schroedinger Eigenmaps

Non-linear transformation methods are based on non-linear manifold learning algorithms, some of which are described in the previous section. These approaches use a graphical model to represent the geometric structure of the data and then by using non-linear manifold learning algorithms, such as LE or SE, the data is transformed to a lower dimensional space. According to the graph model, the vertices in the graph represent the pixels in the image and the edges connecting vertices represent the spectral similarity between pixels. For both LE and SE, the eigendecomposition of the respective operators (described below) provides the basis of the data in the new space. Since SE approach uses a graphical model to represent the local structure of the data, some terminology associated with it is discussed first. Therefore, basic graph theory notation and a brief discussion about the k nearest neighbor (knn) graph and its variation are introduced in Sec. 2.1. Section 2.2 introduces the Graph Laplacian, and its eigendecomposition. Section 2.3 presents the Schroedinger operator, the algorithm to compute it, and to perform its eigendecomposition.

2.1 Graph Terminology, knn Graph and its variations

A graph is a diagram that shows the relation among variables in some feature space by means of connections between them. Therefore, a graph $\mathbf{G} = (V, E)$ is mathematically defined as a set of *vertices* or *nodes*, $V = (v_1, ..., v_m)$, and *edges*, $E = \{e_{ij}\}$, that represents the relationship between vertices.[30] For example, if there is some relation between nodes v_i and v_j, there must be an edge e_{ij} in the graph $\mathbf{G}$ connecting them. There are two kinds of graphs, *weighted* and *unweighted*. All edges in an unweighted graph are equivalent, while each edge in a weighted graph has an associated value, i.e. a weight w_{ij}. There is also another categorization for graphs, *directed* and *undirected*. The edges in a directed graph have a direction associated so that the order of the pair of vertices is relevant. In undirected graphs, the order of the vertices connected by an edge is irrelevant, so the edges are symmetric, that is $e_{ij} \in E$ implies that $e_{ji} \in E$. Then, an undirected and weighted graph will have $w_{ij} = w_{ji}$. Here we only consider undirected graphs.

In the representation of the graph, the vertices are represented by numbers $1, 2, 3, ..., m$ where

m is the number of pixels in the image, and the edges are stored in an $m \times m$ matrix $\mathbf{A}$, known as the *adjacency matrix* that represents the connectivity of the graph, i.e. $\mathbf{A}_{ij} = 1$ if v_i is connected to v_j, and $\mathbf{A}_{ij} = 0$ otherwise. There is another matrix that also represents the connectivity in the graph, but that it is used for weighted graphs. The matrix is the *weight matrix* $\mathbf{W}$, whose entries are the weights of the associated edge, and if there is not an edge connecting two nodes, the entry will be 0. In our case, we assume that the vertices are not connected to themselves, so the elements on the main diagonal of $\mathbf{A}$ and $\mathbf{W}$ are all zero, $a_{ii} = 0$, $w_{ii} = 0$.[30] Therefore, the weights represent how strong the connection is between two vertices, so if two vertices, v_i and v_j are similar, they will have an edge connecting them with a high magnitude of weight, w_{ij}. Given the close relation between the Laplace-Beltrami operator and the heat flow equation, a good option to compute the weights in a graph is the heat kernel. Therefore, the weight function based on the heat kernel is given by

$$\mathbf{W}_{ij} = \begin{cases} \mathrm{e}^{-\frac{d(i,j)^2}{\sigma^2}} & \text{if } \mathbf{G}_{ij} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $d(i,j)$ is the pairwise distance, σ is a parameter estimated or set by the user. In this case, a self-tuning method to estimate σ is used and described in detail in Ref.[31] Regarding the $d(i,j)$ term, we use two pairwise distances in the spectral space, the *Euclidean distance* (ED) and the *Spectral Angle distance* (SAD). Although, ED is the most common metric used in these approaches, Spectral Angle is introduced here given its invariance to linear scale changes.

The graph model as a first step in the process of transforming the data set, $\mathbf{x} = \mathbf{x}_1, ..., \mathbf{x}_m$ in $\mathbb{R}^N$, needs to exemplify the local neighborhood relationship between the data points. There are different techniques to represent this connectivity and to build the graph. The techniques are derived from thresholding processes or by connecting a data point with its k nearest neighbors. The most widely used are: *the ϵ-neighborhood graph*, *the k-nearest neighbor graph*, *the fully connected graph*,[32] where each vertex is connected to all other vertices in the graph, and *the Mutual k-nearest neighbor graph*[33] that presents an edge connecting two vertices if each of them is a k neighbor of the other. Here, the k nearest neighbor (knn) graph is used along with an adaptive method to estimate k that is based on the local spectral density of the each data point.

2.1.1 knn Graph

This is the most common and straightforward technique used in graph theory-based algorithms. In this case, two nodes v_i and v_j are connected if one of them is among the k nearest neighbors of the other. The number k is commonly chosen arbitrarily, which is its main disadvantage. Figure 2 shows a synthetic data set and its graphs with two different values of k. The data set was generated as a sum of points drawn from normal distributions in $2D$ with different means and the same covariance and consists of two classes near-linearly separable and differentiated by color (see Fig. 2a). Figures 2b and 2c show the graphs with $k = 2$ and $k = 8$ built on the data set. Obviously, when k is increased, the number of edges in the graph also increases, so the choice of a correct number for k becomes a trade-off problem. A large k allows us to keep some important connections in the graph that may resemble the local structure of the data, but for small k the $\mathbf{A}$ and $\mathbf{W}$ matrices are more sparse leading to computational efficiency. Another important remark to have into consideration is the number of components of a graph, which at one point could give insight to the data structure and of how many classes are presented in the scene. For example for the data set in Fig. 2a, the graph has 8 components with $k = 2$, which may be an over partitioning

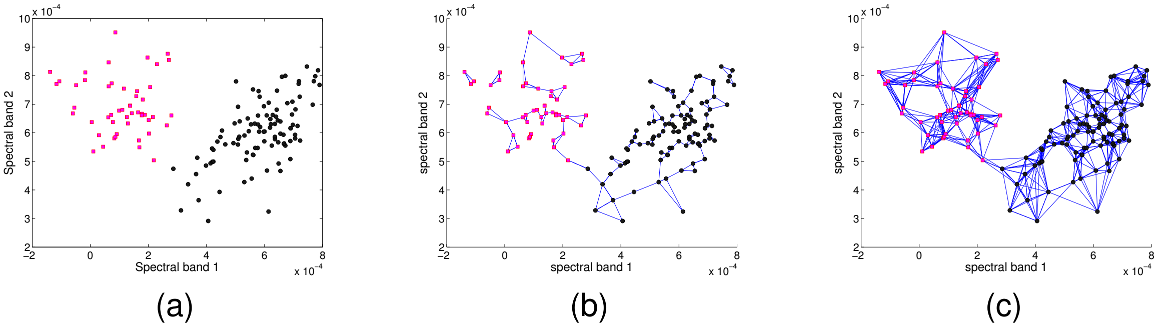


Fig 2 2-D Synthetic data set. (a) Original data set is a sum of Gaussians with different means and equal covariances. The data set consists of two classes differentiated by color. (b) knn Graph with $k = 2$. (c) knn Graph with $k = 8$

given the data set has two classes, while for $k = 8$, the graph has only 1 component. Thus, we could conclude that an appropriate number for k would be between 2 and 8, but how to estimate this number is again a trade-off problem.

In order to compensate for its disadvantages, adaptive methods to estimate k have been proposed. For example, a method that attempts to assign different values of k to each data point is presented in Roweis and Saul (2000).[16] This method assigns a different value of k to each pixel in the data set based on counting how many times each pixel is a neighbor of the rest. This counting is made sequentially and stops when all data points have at least been assigned once as a neighbor. The problem with this method is that one outlier in the data can drive high k values resulting in an over connected graph, which would imply a high computational cost and a mis-representation of the local data structure.

2.1.2 Weighted density knn estimation

This is another technique to estimate k in a locally adaptive way[25] with the goal of minimizing the impact of pixels outside clusters of the data. The general idea of this method is to assign few neighbors to those pixels in regions of the spectral domain with the lowest density, and a maximum k neighbors (k_{max}) to those with the highest density. In this way, a spectral co-density

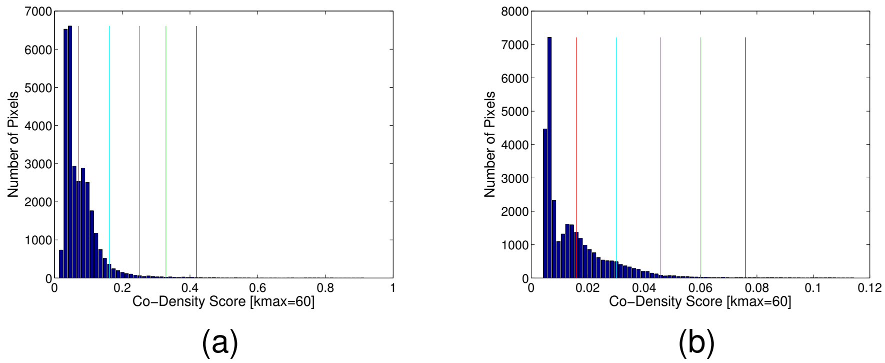


Fig 3 Histogram of Co-density score calculated for the Hyperspectral Image shown in Fig.9b. (a) using Euclidean Distance. (b) using Spectral Angle Distance

score is computed for each pixel and based on it, a k value is assigned. The co-density score is computed as the sum of the distances from each pixel to its k_{max} neighbors divided by k_{max}, where k_{max} is set by the user and usually is a high value (e.g. 60). Then, the distribution of co-density measurements over the data sets is divided into segments and the pixels that fall into a same segment will be assigned the same k value, which will be different from the k values for other portions of the distribution. According to Mercovich, et.al. (2011) and from experimental results, the co-density distribution in HSI is generally well represented by a normal distribution and here, we divide the distribution into 6 partitions. The partitions are defined by segmenting the distribution based on the standard z-scores, where the lowest threshold is set to the z-score of -2 and the highest to z-score of 2, and the values for k are defined experimentally. In our case, the k values are defined in terms of the k_{max} times the minimum value of the co-density score over the maximum co-density value in each partition. This computation was chosen considering a constant variation for k through the segments.

Figures 3a and 3b show two co-density histograms for two different distances, ED and SAD, computed for a hyperspectral image. The figures also show the partitions in which the histograms are divided.

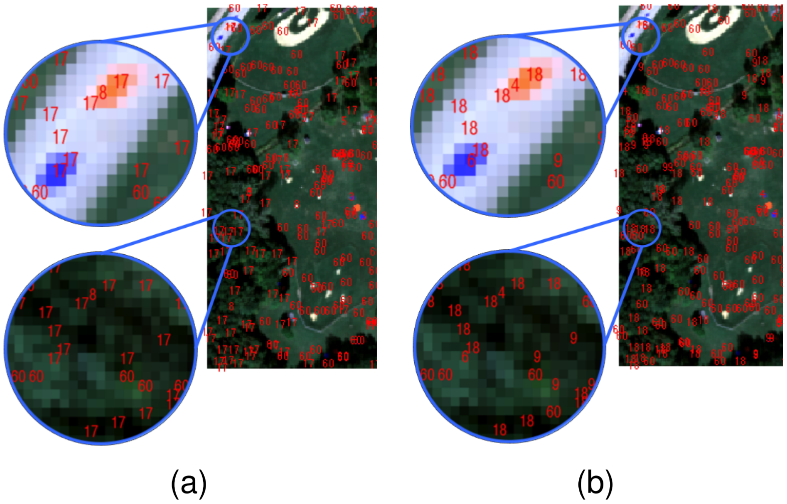


Fig 4 RGB version of a Hyperspectral Image with the k number for different pixels. The knn was estimated by weighted density knn method using: (a) ED. (b) SAD. Each figure has a close-up of two areas with different variability that is reflected in the estimated knn.

For SAD and ED, there is not much difference regarding the co-density distributions, obviously there is difference in the value ranges and the histogram for SAD is wider (in a z-score sense), so in this case, there will be more pixels in each segment sharing the same value of k. Regions with less variability tend to be in high density regions such as the open field, and these regions have pixels with the highest value of k ($k_{max} = 40$). Regions with more variability such as the forest have different values for k. Likewise, brightest pixels or those pixels on the target panels have smaller values of k. Figures 4a and 4b show the k values assigned to various pixels on the image, which were randomly chosen. As shown in the figures, the k values are varied and in general the trend of values match with the original idea.

Returning to the example shown in Fig. 2, Fig. 5b shows the graph built using the weighted density method with ED. The synthetic data is shown again in Fig. 5a, and the knn graph with $k_{max} = 8$ in Fig. 5c. It can be observed in Fig. 5b that regions with high density are more interconnected, i.e. there are more edges connecting the points, and regions with low density such as points at the periphery of the two clusters are connected to only one other point in the graph,

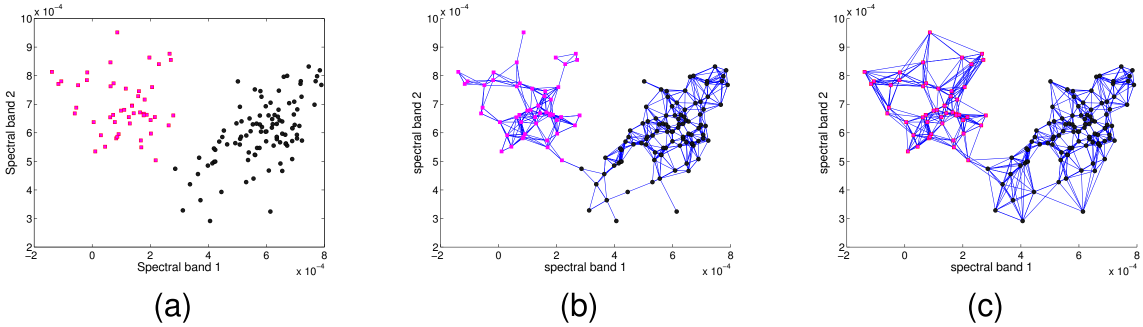


Fig 5 2-D Synthetic data set. (a) Original data set is a sum of Gaussians with different means and equal covariances. The data set consists of two classes differentiated by color. (b) Adaptive knn Graph by using the weighted density knn estimation. (c) knn Graph with $k = 8$

i.e. those points have few connections. In this case, although the number of components does not match with the number of classes in the dataset, the result is not too bad since only one edge is connecting the 2 clusters. From a clustering point of view, a more complex process may be necessary, for example a LE or SE analysis.

2.2 Graph Laplacian and its Eigendecomposition

The eigendecomposition of regular and symmetric graphs has been of interest to the spectral graph community given the close relation between the eigenvalues and all major invariants of a graph.[2] Specifically, and from the study of the set of eigenvalues, i.e., the spectrum of a graph, it is possible to deduce the properties or structure of the graph. Likewise, the set of eigenvectors are used as a representation of the data in the new space. When the data are projected into the new space using the eigenvectors, the new data are termed eigenmaps. In this way, the eigenvectors of the Laplacian have had a wide application. the Laplacian operator gives information about the local connectivity or relation between neighboring pixels. Therefore, Laplacian eigenmaps as a nonlinear local transformation, assumes that the data set $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$ in $\mathbb{R}^N$ lies on a $l-$manifold, where the dimension l is lower than N. The main goal is then to find the representation $\mathbf{y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_m\}$

of the data set in the Laplacian space.[6] As a first step, the HSI data that can be seen as a cube with two spatial dimensions and one spectral dimension is organized as a matrix of $[m \times n]$ size, with m as the total number of pixels in the image and n the spectral dimension. Then, the HSI matrix is used as input to the process that computes the Laplacian operator and its eigenmaps. This is analogous to the standard Principal Components transformation, however using the decomposition of the graph operator instead of the spectral covariance matrix.

The algorithm is stated as follow:

1. *Graph construction.* The similarity matrix $\mathbf{G}$, which represents the connectivity of the data set as it was presented in Sec. 2.1, is built by using the knn graph method or the weighted density method and the weights are assigned by following Eq. (1), with Euclidean or Spectral Angle as metric distances $d(i, j)$. Given the way in which the input data is sorted, the weight matrix and the graph Laplacian operator are $[m \times m]$ matrices.

2. *Computation of Laplacian operator.* Once the weight matrix $\mathbf{W}$ is computed, the Laplacian operator of the graph is calculated by following:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \tag{2}$$

   with $\mathbf{D}$ as a diagonal matrix whose entries are given by the sum of either rows or columns of the weight matrix, $\mathbf{D}_{ii} = \sum_{j=1}^{m} \mathbf{W}_{ij}$ (since the weight matrix is symmetric). $\mathbf{D}$ is termed the weighted degree matrix.

3. *Solving the eigenvalue problem.* The final step is to find a representation that preserves as best as possible the local structure of the data, i.e. nearby connected points, $\mathbf{x}_i$ and $\mathbf{x}_j$ in $\mathbb{R}^N$

stay as close together as possible after the transformation. Therefore, a penalty is imposed if $\mathbf{x}_i$ and $\mathbf{x}_j$ are mapped far apart. Considering the mapping $\mathbf{y} = \{\mathbf{y}_1, \mathbf{y}_2, .., \mathbf{y}_m\}$ in the new l-dimensional Laplacian space and given the choice of weights, a good option is to minimize the object function $(\mathbf{y}_i - \mathbf{y}_j)^2\mathbf{W}_{ij}$, where for any $\mathbf{y}$, it is true (for a detailed demonstration see[26]):

$$\min_{\mathbf{y}^T\mathbf{D}\mathbf{y}=\mathbf{I}} \frac{1}{2}\sum_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2\mathbf{W}_{ij} = \min_{\mathbf{y}^T\mathbf{D}\mathbf{y}=\mathbf{I}} \text{trace}(\mathbf{y}^T\mathbf{L}\mathbf{y}) \tag{3}$$

where $\mathbf{I}$ is the identity matrix. This minimization problem is equivalent to finding the l minimal solutions of the eigenproblem $\mathbf{L}\phi = \lambda\mathbf{D}\phi$, that in matrix form is written as $\mathbf{L}\Phi = \Phi\mathbf{D}\Lambda$. Then, the eigenvectors sorted by columns, corresponding to the l smallest non-zero eigenvalues, are the vector basis in the Laplacian space. The trivial solutions, i.e., the zero eigenvalues, are not considered.

2.3 The Schroedinger Operator on Graphs and its Eigendecomposition

The Schroedinger Operator and the use of its eigenmaps in classification problems has been proposed in Czaja and Ehler (2013)[7] and Benedetto et.al. (2012).[8] They proposed a classification framework that allows the use of labeled data for improving classification performance, where the effect of a "potential term" is applied not just to the labeled points, but to the pixels that are spectrally similar to them regardless of their spatial position. The potential term is included in the definition of Schroedinger operator and is described below. One type of potential is the *barrier potential*, whose effect is to influence the diffusion process on graphs, so that the labeled points are pulled toward the origin in the Schroedinger Space. Therefore, by appropriately choosing the locations of the barrier potentials, it is possible to identify and separate a specific cluster that contains the labeled data points from the other pixels in the image.

Since the Schroedinger operator can be seen as a Laplacian with potentials, the Schroedinger operator is a generalization of the Laplacian. The algorithm for obtaining the embedding in the Schroedinger space follows the same LE algorithm, where the final goal is to perform the eigendecomposition of the Schroedinger operator instead of the Laplacian. In this way, the first two steps presented in the previous section are kept and performed the same way as they were described. Following that, the Schroedinger operator and its eigendecomposition are computed. The Schroedinger operator is given by

$$\mathbf{S} = \mathbf{L} + \alpha \mathbf{V} \tag{4}$$

where α is a parameter that allows emphasizing the significance of $\mathbf{V}$ with respect to $\mathbf{L}$, and the barrier potential, $\mathbf{V}$, is a $[m \times m]$ nonnegative diagonal matrix encoding the location of the pixels labeled as the target class. Basically, the labeled pixels in the potential term are constraints imposed in the embedding that "pushes" them go toward the origin in the Schroedinger space; how close these points are from zero will depend on the value of α. Once the Schroedinger operator is computed, the next step is to find the embedding that is given by minimizing Eq. (3) but with the Schroedinger operator instead of the Laplacian. Thus, here it is solved as

$$\min_{\mathbf{y}^T\mathbf{D}\mathbf{y}=\mathbf{I}} \text{trace}(\mathbf{y}^T(\mathbf{L} + \alpha\mathbf{V})\mathbf{y}). \tag{5}$$

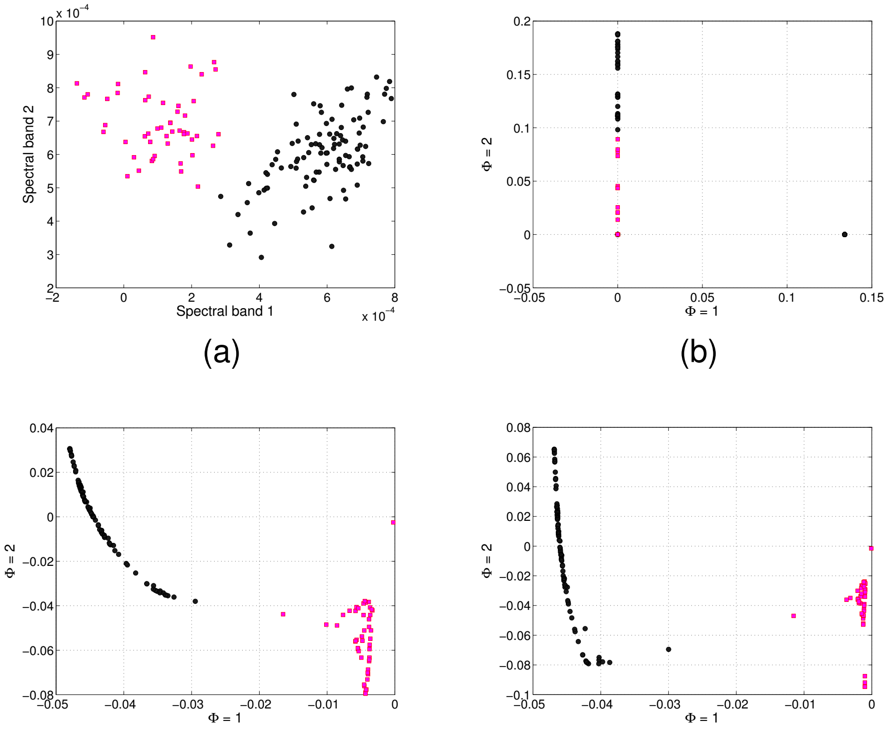
Equation (5) could be rewritten as

$$\min_{\mathbf{y}^T\mathbf{D}\mathbf{y}=\mathbf{I}} \frac{1}{2} \sum_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2 \mathbf{W}_{ij} + \alpha \sum_i \mathbf{V}_i \|\mathbf{y}_i\|^2. \tag{6}$$

The first term in Eq. (6) corresponds to the Laplacian operator and the second term to the barrier potential. Analyzing with more detail this second term, if α tends to 0 SE will tend to LE. For a different α value, the diagonal potentials introduce a penalty at each nonzero entry which creates a well at the corresponding node, i.e. if $\mathbf{V}_i \neq 0$, where i corresponds to the spatial location of the target pixels in the image, then $\|\mathbf{y}_i\|^2 \leq \frac{C}{\alpha \mathbf{V}_i}$ with C as a constant and, as $\alpha \to \infty$, $\mathbf{y}_i \to 0$. This means that as soon as α increases, the labeled pixels in $\mathbf{V}$ are pulled toward zero while the embedding is performed, and because the embedding must preserve the local structure of the data, the neighbors of those pixels (as connected via the graph $\mathbf{G}$) noted as target pixels are also pulled toward zero.[8] This trend continues until a stabilization point is reached and no matter how large α is, the embedding does not steer anymore in the direction set by the labeled points. Finally, the mapping $\mathbf{y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_m\}$ as in LE, is computed by solving the eigenproblem $\mathbf{S}\Phi = \mathbf{D}\Phi\Lambda$ and the l eigenvectors corresponding to the first l smallest eigenvalues are the basis of the representation in the Schroedinger space.

The following figures show the influence of α and k in the Schroedinger embedding. Recalling the example of the two classes with a combination of normal distributions, Fig. 6 shows the representation of the original data in the Schroedinger space for three different approaches where the number of k has been varied and α was arbitrarily set to 100 for all three cases. The class in magenta color has been picked as the target class and just one of its points was chosen as a labeled data point in the potential $\mathbf{V}$. In all cases, the target class is pulled toward the origin. Obviously there are some points closer than other ones, representing the level of similarity between each point and the labeled one. Regarding the separability between the two classes, $k = 2$ looks like the most difficult case, while for the other two, $k = 8$ and the adaptive k, the target class is well separated, so it will be easy to set a threshold to detect it.

Continuing with the same example and varying now α, Fig. 7 shows the first two LEs and



(a) (b)

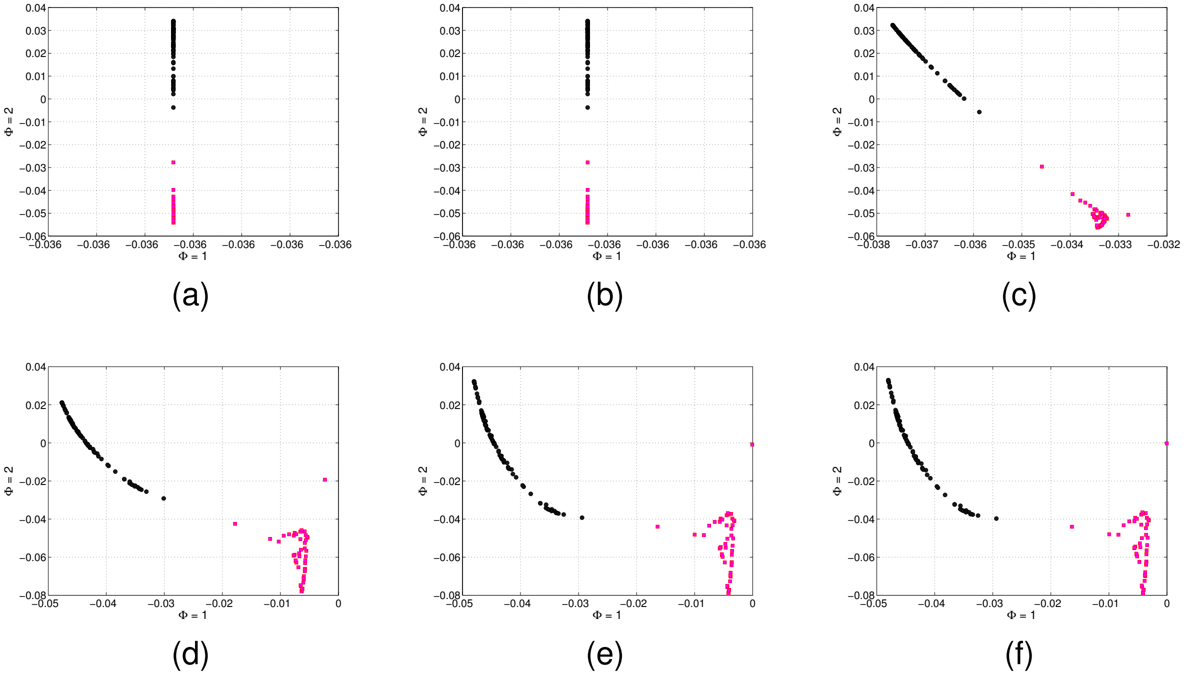
(c) (d)

Fig 6 2-D Synthetic data set. (a) Original data set is a sum of Gaussians with different means and equal covariances. The data set consists of two classes differentiated by color. (b) SE of the data set with $k = 2$. (c) SE of the data set with $k = 8$. (d) SE of the data set using weighted density knn

the SEs at three different values for α. As it was described before, when $\alpha = 0$, the Schroedinger embedding is the same as the Laplacian embedding. As α increases the pixels in magenta, which is the class labeled in $\mathbf{V}$, are pulled toward the origin until it reaches a balance point. This is shown in the last two plots, the two largest α values, where there is not much difference in the embeddings.

3 Detection using SE: Methodology and detector definition

As it has been established above, the most suitable characteristic of SE method is that the position in the Schroedinger space of the target pixels labeled in $\mathbf{V}$ and those similar to them, is predetermined and known, i.e. non-zero labels in $\mathbf{V}$, are penalties that “push” the pixels toward the origin in Schroedinger space. This fact is taken into consideration to define a detector that penalizes pixels that are far from zero and assigns high values to pixels close to zero. Mathematically, the



(a) (b) (c)

(d) (e) (f)

Fig 7 Scatterplots for different α values. The scatterplots include the first two smallest eigenvectors as axis, (a) LE, (b) SE with $\alpha = 0$, (c) SE with $\alpha = 0.1$, (d) SE with $\alpha = 10$, (e) SE with $\alpha = 300$ and (f) SE with $\alpha = 1000$.

detector is expressed as

$$\mathbf{T}_{SE_i} = \frac{1}{\|\Phi_{SE_i}\|} \tag{7}$$

where Φ_{SE_i} are the Schroedinger eigenmaps for each pixel i in the l dimensional space, and $\|.\|$ is the Euclidean norm. Thus $\|\Phi_{SE_i}\|$ is the magnitude of each pixel vector i after transformation to the Schroedinger space. As was shown in the previous Section, the α parameter has an impact over the Schroedinger embedding and particularly over the separation between the labeled pixels (the target pixels), and the background pixels. Because α is a data-dependent parameter, the parameter $\hat{\alpha}$ is introduced as a scaling parameter that does not depend on each data set. We define, $\alpha = \hat{\alpha}(\mathrm{tr}(\mathbf{L})/\mathrm{tr}(\mathbf{V}))$.[34] Figure 8 shows detection maps when the red panel in the scene is detected at different $\hat{\alpha}$ values. One pixel from the red panel is picked as target pixel and is label in the $\mathbf{V}$ matrix. The trend of detection maps vs. $\hat{\alpha}$ is very similar as the trend in SEs vs. α shown in Fig.

7. As α is increased, there are more target pixels being detected.

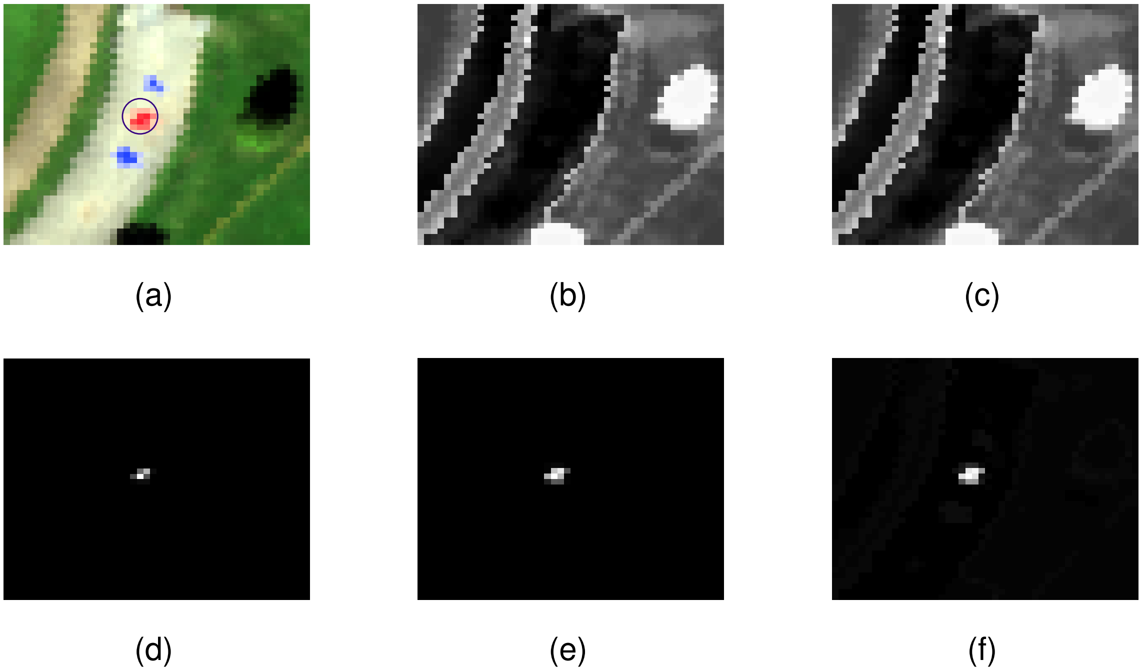


Fig 8 Detection maps for different $\hat{\alpha}$ values. (a) RGB rendering of the scene with the target panel framed by a circle. Detection maps were computed using the first two smallest: (b) LE, (c) SE with $\hat{\alpha} = 0$, (d) $\hat{\alpha} = 0.06$, (e) SE with $\hat{\alpha} = 23$ and (f) SE with $\hat{\alpha} = 100$.

The detector is the final step in the process. However, the overall detection process includes some previous steps that are based on the algorithm described in Sec. 2.2. The detection methodology specifically proposed here, is as follows:

1. **Graph creation - weights computation.** The graph is built by following either of the two techniques described in Sec. 2.1.1, the graph knn and the density weighted graph. These techniques are used to estimate the number of k nearest neighbors to which each pixel has to be connected. In addition, and for improving the connectivity of the target with pixels which share some similarity but not enough to be directly connected, the target neighborhood is defined as the pixels that are neighbors of the target with connection length $<= 2$, i.e., the target pixel is connected to its neighbors initially found by knn graph or density weighted

graph, and also to the neighbors of its neighbors. Once the graph connectivity is built, the weights are computed by using Eq. (1). Two different methods for defining the target signal are described below.

2. **Schroedinger Operator.** Before computing the Schroedinger operator, the matrix $\mathbf{V}$ must be defined. In this case, the barrier potential matrix is used. Therefore, $\mathbf{V}$ is a diagonal matrix with labels along the main diagonal at the position of the target pixel and its neighbors (obviously considering the definition of target neighborhood presented in the previous step). The labels in $\mathbf{V}$ are represented by 1s along the diagonal. Finally and after computing the Laplacian operator with Eq. (2), the Schroedinger operator is calculated by using Eq. (4).

3. **Eigendecomposition.** Once the Schroedinger operator is computed, its eigendecomposition is performed and only a given l eigenvectors are kept as basis vectors in the Schroedinger space. The number l, i.e., the dimension in the Schroedinger space, can be estimated or set experimentally.

4. **Detection.** With the l first SEs and using Eq. (7), a detection score is computed for each pixel. The combination of all the detection scores form a detection map image, where higher values represent a higher likelihood that the pixel belongs to the target class.

4 Experimental Results

As has been established above, SE has the property of steering the transformation in some convenient direction favoring a separation between the labeled target class and the background without the necessity of knowing the background characteristics. The methodology based on SE for target detection that is presented here, includes a qualitative analysis and a quantitative assessment of two approaches, *in-scene pixel detection* and *target spectral signature detection* that are described below. The experiments include the use of knn Graph and Adaptive knn Graph methods for building the initial graph. For the knn Graph method, two values for k are used, $k = 4$ and $k = 20$. The weighted density estimation method described in Sec. 2.1.2 was used as the adaptive knn Graph method (Adp$k$) with a $k_{max} = 40$. For each approach, in-scene detection and target signature detection, there are six different results, including the variations of the two similarity distances, ED and SAD. The number of SE used in Eq. (7) to compute the detection map was estimated by the Gram matrix method, a method based on geometric models developed for estimating the spectral complexity of an image,[35] but also has been used to estimate the number of pure materials that constitute an image.[36] An experiment to set the value of $\hat{\alpha}$ was previously performed and based on it, $\hat{\alpha}$ was set to 0.11 (see Ref.[34] for more details about the experiment). In addition, the ACE detector, as one of the most used target detection methods, is also used in order to compare and assess the SE approach.

4.1 Experimental Data

Reflectance spectral data sets are used in this work in order to assess the application of SE in target detection. The data sets were collected as part of two data campaigns, SHARE 2010 and SHARE 2012, hosted by Rochester Institute of Technology (RIT). The SHARE 2010 scene considered in this work covers part of the RIT campus.[37] The urban scene contains several materials or objects such as asphalt, brick, vegetation, concrete, roof, buildings, cars, parking lots, etc. Blue and red cotton felt panels in 2m ×2m and 3m ×3m sizes are also included. The panels are located at areas with different illumination and occlusion conditions. The image has 86×146 pixels and 328 bands after absorption bands are removed. The SHARE 2012 campaign, in turn, was conducted in and around Avon, NY, in September 2012.[38] The scene considered here is an image of 250×123 pixels and 392 spectral bands after removing the absorption bands. The scene is primarily covered by vegetation and there are various felt panels in red and blue colors, some clearly visible and others covered by shadows cast by surrounding trees. The exact actual blue and red cotton felt panels were used in both data campaigns, and the same hyperspectral spectrometer *ProSpecTIR VS* was also used to capture the hyperspectral images. The sensor covers the Visual, NIR and SWIR $(0.4 - 2.4\mu m)$ ranges with 360 bands and has a spatial resolution of approximately 1m. Figure 9 shows these data sets, the two scenes, SHARE 2010 and SHARE 2012, and the blue material spectral signature that is used as target. The panels of interest will be the blue panels framed in yellow squares and magenta circle in SHARE 2010 and 2012 scenes shown in Figs. 9a and 9b. These blue panels in both scenes are cotton felt panels that belongs to the same material, the only difference among them is their size. The SHARE 2012 campaign also has available multiple target spectral signatures of the blue and red panels measured in the lab and in the field. Figure 9c shows the target spectral signature of one of these panels measured in the open field, which is used as target signature for both scenes, SHARE 2010 and SHARE 2012.

As to the dimensionality of each considered data set, Gram matrix method that was proposed to estimate pixels containing pure materials in an image[36] was used to estimate the dimensionality of SHARE 2010 and SHARE 2012 in reflectance domain. Hence, the dimensionality for SHARE 2010 is 12 and for SHARE 2012 is 10.

4.2 Target detection - Qualitative Results

The SE algorithm for target detection described in Sec. 3 was implemented in MATLAB program. The ATRIA tool which is implemented in C$++$ was used for building the knn graph.[39] A *mex* file

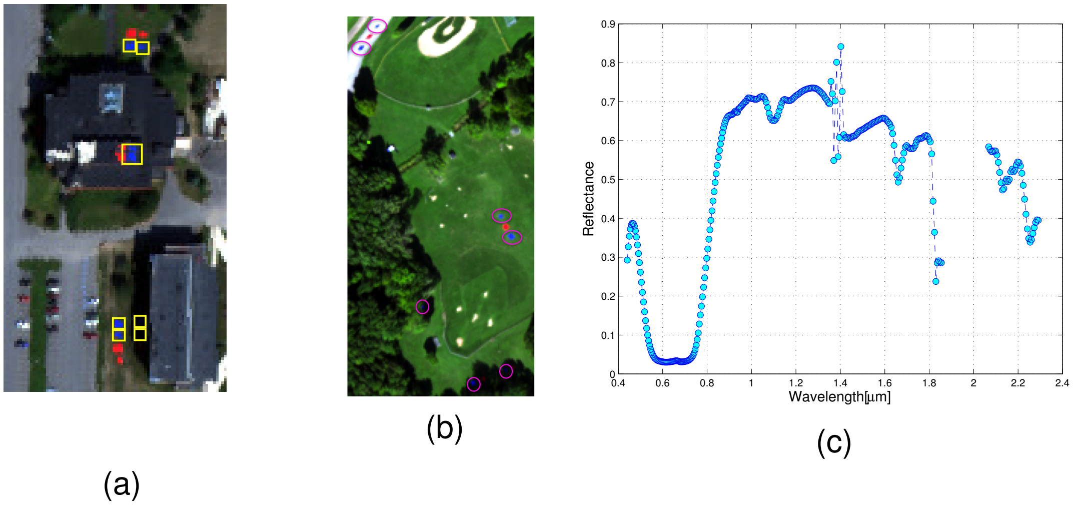


Fig 9 Hyperspectral Data Sets used in this paper. (a) RGB rendering of SHARE 2010 hyperspectral image with target locations highlighted. (b) RGB rendering of SHARE 2012 hyperspectral image with target locations highlighted. (c) Spectral signature of blue felt target measured in the field.

was generated in order to use ATRIA from MATLAB. In addition, MATLAB built-in function *eigs* was used to solve the eigenproblem. The run time depends on the approach used for building the graph, roughly spanning from 8 minutes for knn Graph with $k = 4$ to 44 minutes for the Adp$k$ and $k = 20$ methods as run on an AMD computer with 1.4 GHz.

4.2.1 In-scene Pixel Target

In this approach, the target class is represented by a set of pixels taken from the image. This in some way ensures that there are actual pixels belonging to the class of interest in the scene and in turn, there is a good match with the true targets. Figure 10 shows the flowchart for this approach. In this case, a pixel is picked from one of the blue panels in the image and is designed as the target pixel. Then, that pixel and its spectral neighbors are labeled with 1s in $\mathbf{V}$. The input data is then the hyperspectral image represented as an $[m \times n]$ matrix.

Fig 10 Flowchart of the approach where an in-scene pixel and its neighbors are used as target, i.e. are labeled in $\mathbf{V}$. The input image data is a $[m \times n]$ matrix, where m is the number of pixels and n the number of spectral bands.

The qualitative results for SHARE 2010 and 2012 include detection maps obtained after applying the approaches based on SE and its binary versions that are shown in Figs. 11 and 12. These images include the detection maps (left) and a thresholded version (right) for each considered SE approach. The threshold value was experimentally set to 1000 for all results. Results for ED are shown in the first two columns. Each row starting from the top corresponds to $k = 4$, $k = 20$ and adaptive knn respectively. The same organization is followed for SAD results shown in the last two columns. **SHARE 2010 results** show for all the cases that only some of the 5 blue panels in open field are detected and the other 2 blue panels that are in the shadows are never identified. The $k = 20$ approach with SAD and ED can identify 5 out of 7 panels as well as the Adp$k$ with SAD.

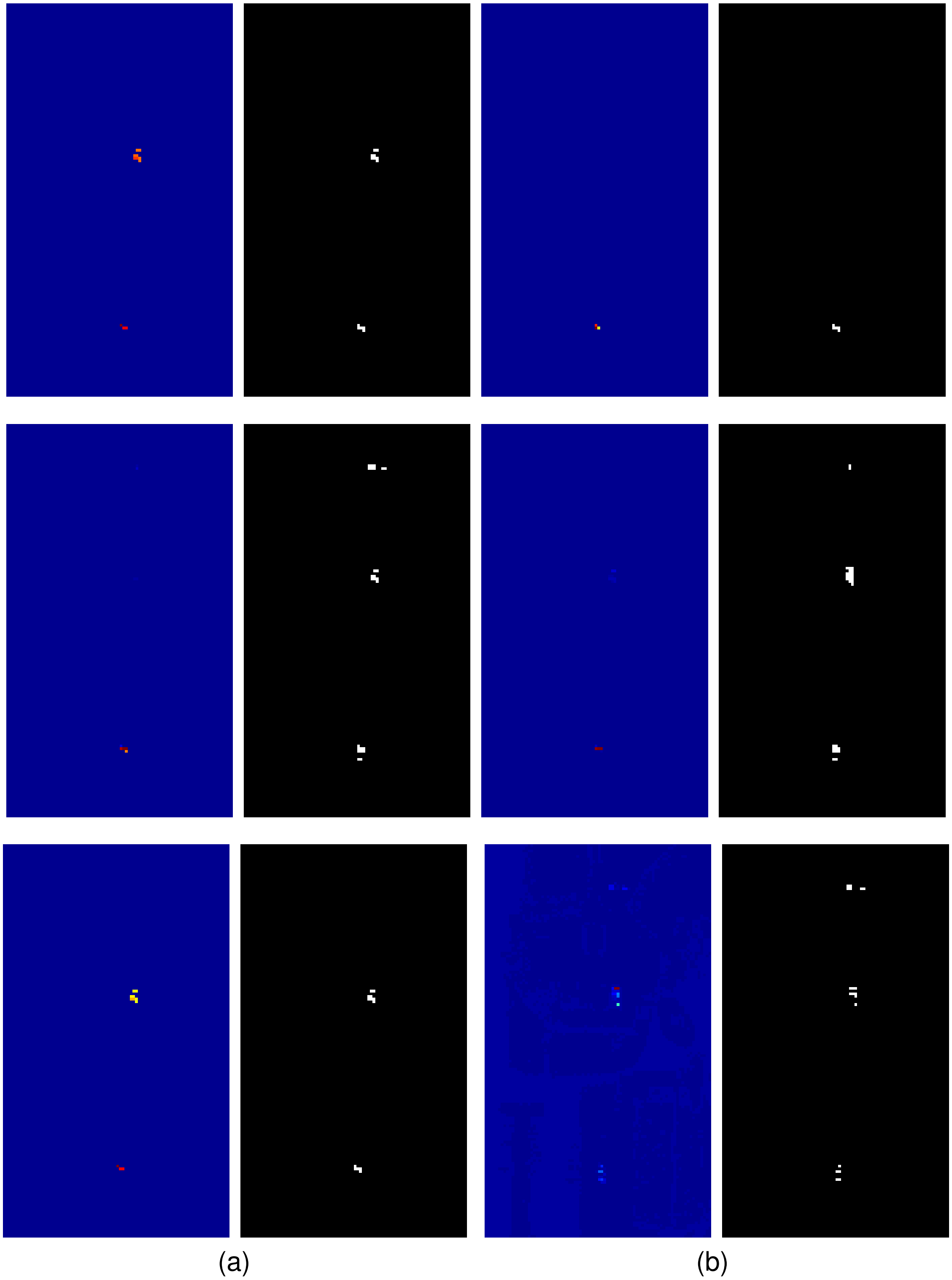


Fig 11 Detection map (left) and binary image (right) for the *In-scene target detection* approach in SHARE 2010. (a) using Euclidean Distance. (b) using Spectral Angle Distance. The first row shows results for $knn = 4$, second row are the results for $knn = 20$ and the third row corresponds to results for Adaptive knn.

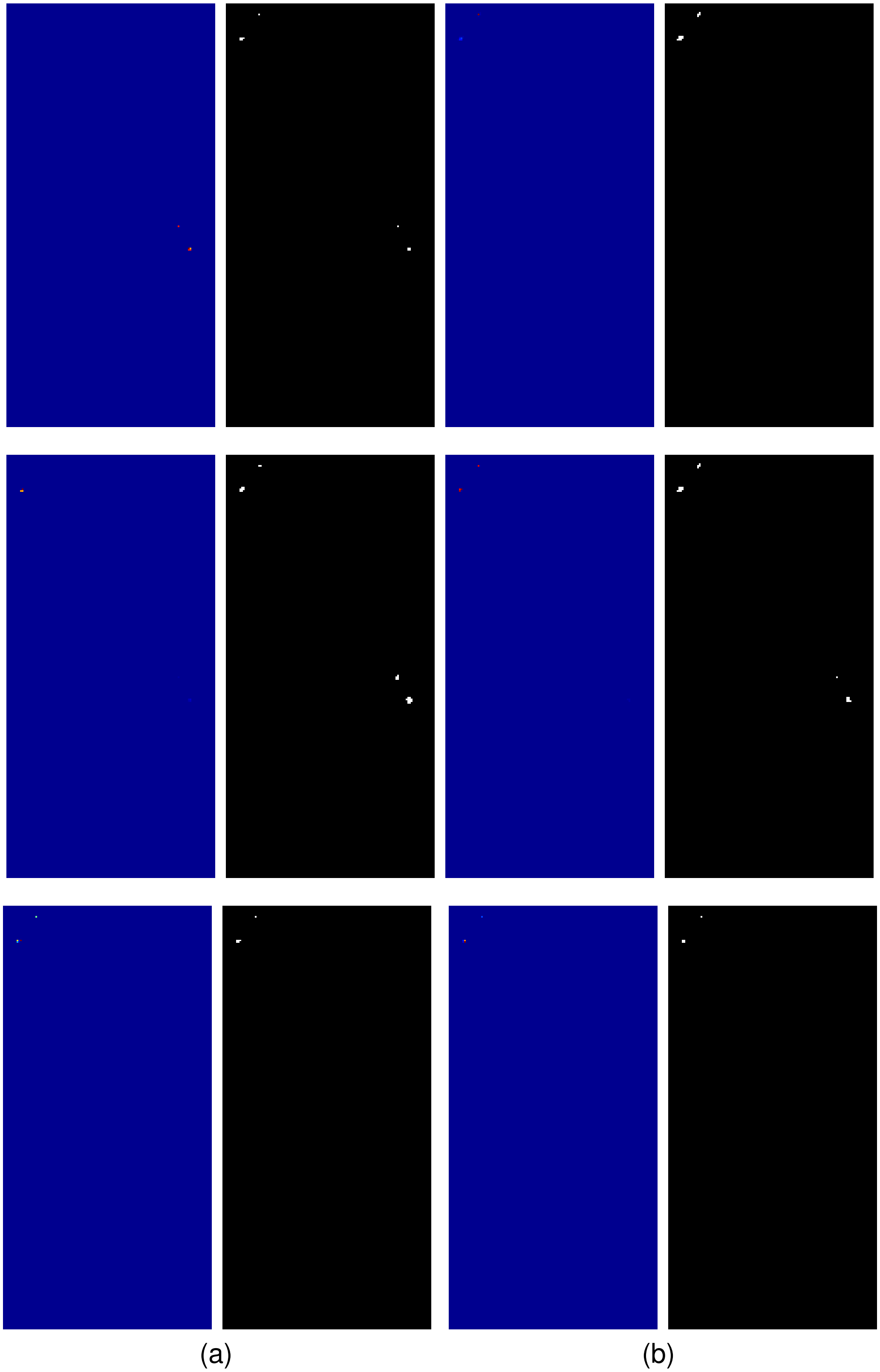


Fig 12 Detection map (left) and binary image (right) for the *In-scene target detection* approach in SHARE 2012. (a) using Euclidean Distance. (b) using Spectral Angle Distance. The first row shows results for $knn = 4$, second row are the results for $knn = 20$ and the third row corresponds to results for Adaptive knn.

SHARE 2012 results are shown in Fig. 12, including results for ED in 12a and for SAD in 12b. Visually, there is not much difference in the number of blue panels found by the different approaches. In the best case only 4 out of 7 blue panels are identified. These four panels correspond to the panels in the open field, i.e. the panels most exposed and panels easiest to find. The three panels in shadows are completely missed.

4.2.2 Field Target Spectral Signature

In this case, the spectrum measured in the field of one of the blue target panels is used as the target. This approach has extra difficulties since the target spectral signature does not necessarily have the exact spectral shape of actual target pixels present in the scene due to sensor artifacts and potential errors in the atmospheric compensation process. Figure 13 shows the flowchart for this approach where the spectral signature of the target panel and its neighbors are labeled as target class in $\mathbf{V}$. Therefore, the input is the hyperspectral image plus the target signature, forming a $[m + 1 \times n]$ matrix. The $\mathbf{V}$ matrix is then defined such that 1s are placed at the last entry of $\mathbf{V}$ and at the position of its neighbors.

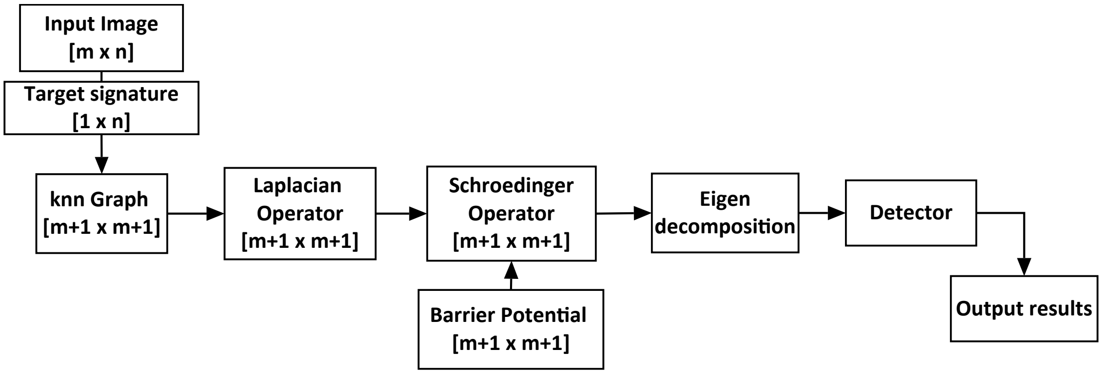


Fig 13 Flowchart of the approach where a target spectral signature and its neighbors are labeled in $\mathbf{V}$. The input data is a $[m+1 \times n]$ matrix, where m is the number of pixels, n the number of spectral bands and $+1$ is the location of the injected target spectrum.

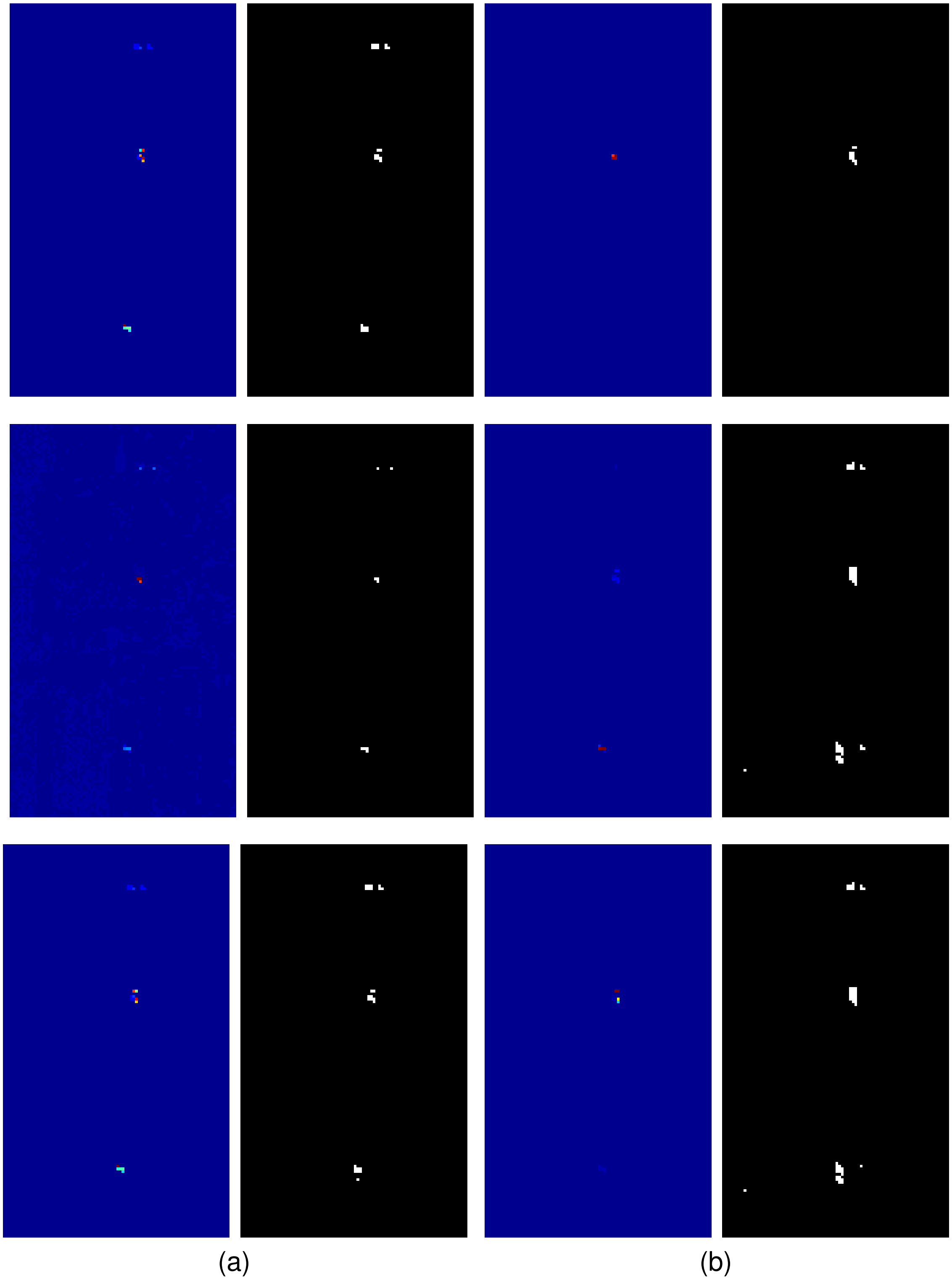


Fig 14 Detection map (left) and binary image (right) for the *Target signature detection* approach in SHARE 2010. (a) using Euclidean Distance. (b) using Spectral Angle Distance. The first row shows results for $knn = 4$, second row are the results for $knn = 20$ and the third row corresponds to results for Adaptive knn.

As in the In-scene detection experiments, the detection maps and their binary images are considered for analyzing the detection performance from a qualitative point of view. The threshold was again experimentally set to 1000 for all the approaches considered, and the results are organized as the results in the In-scene detection.

Results for SHARE 2010 are shown in Fig. 14 using ED (Fig. 14a) and SAD (Fig. 14b). The use of SAD with $k = 20$ and Adp$k$ exhibit the best performances because in both cases one of the blue panels in shadows can be identified, resulting in the detection of 6 out of 7 panels and a few false alarms. For the case of $k = 4$, ED has the best performance in comparison with the use of SAD, although only the five panels in open field are identified. Therefore, the worst performance is obtained for $k = 4$ with SAD.

Regarding **SHARE 2012 results**, the detection maps and the binary images are shown in Fig. 15, including the ED and SAD results in Figs. 15a and 15b respectively. From a visual perspective, SAD distance has a better performance than ED, at least for two of the cases, $k = 4$ and $k = 20$. For the adaptive knn, both ED and SAD distances show the same performance. The best case is obtained when $k = 20$ and SAD are used. In this case, the seven blue panels are identified with a few false alarms. Contrary, the worst performance is exhibit by $k = 20$ and ED approach, only four blue panels are correctly identified but also two red panels are identified as targets.

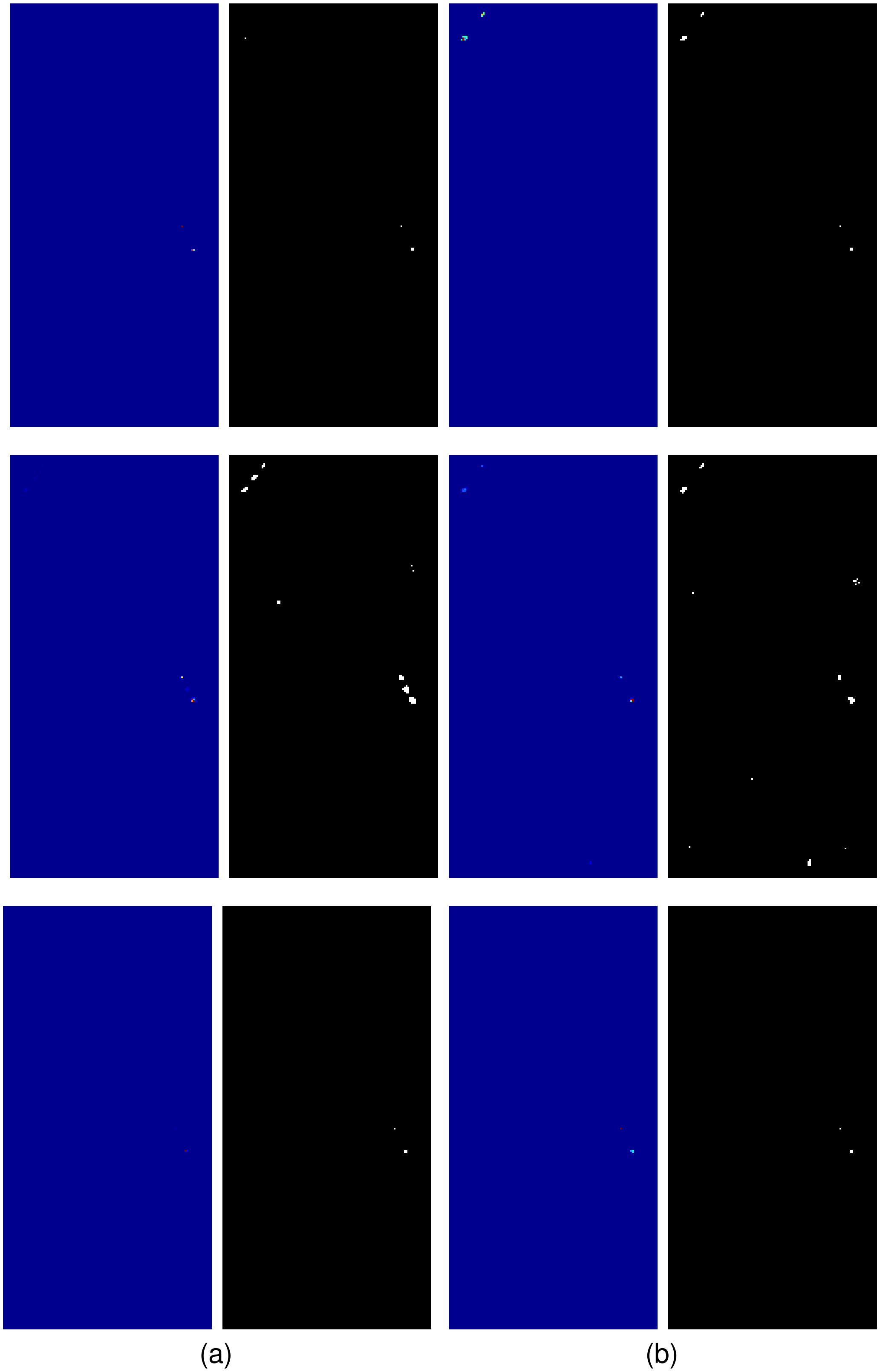


Fig 15 Detection map left) and binary image (right) for the *Target signature detection* approach in SHARE 2012. (a) using Euclidean Distance. (b) using Spectral Angle Distance. The first row shows results for $knn = 4$, second row are the results for $knn = 20$ and the third row corresponds to results for Adaptive knn.

4.2.3 Comparison with ACE detector

In order to see how a typical detector based on statistical or geometric approaches performs on the specific data set used in this paper, the results obtained with the SE methodologies are compared to those obtained with the ACE algorithm.[9] The ACE detector is a common and widely used algorithm based on a statistical description of the data that includes the estimation of the background mean ($\boldsymbol{\mu}$) and covariance ($\boldsymbol{\Sigma}$). The ACE detector is written as:

$$\mathbf{T}_{ACE}(\mathbf{x}_i) = \frac{(\mathbf{t}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i-\boldsymbol{\mu})}{\frac{1}{N}\sqrt{(\mathbf{t}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{t}-\boldsymbol{\mu})}\sqrt{(\mathbf{x}_i-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i-\boldsymbol{\mu})}} \tag{8}$$

where $\mathbf{t}$ is the target signature and $\mathbf{x}_i$ is the actual pixel i under test. Figures 16 and 17 show the results for SHARE 2010 and SHARE 2012, where an in-scene pixel is used as target (Figs. 16a and 17a) and when the target spectral signature is used (Figs. 16b and 17b).

For the in-scene detection, the $\mathbf{t}$ in Eq. (8) is the same pixel as before picked from one of the blue target panels in the image. For **SHARE 2010**, the detection map and its binary version show only the five blue panels in the open field highlighted, while for **SHARE 2012**, four panels also in the open field are completely identified. For the target signature detection, the field-measured spectral signature is used as the target. For both images, SHARE 2010 and SHARE 2012, the detection maps and the binary images show only false alarms. Hence, ACE detector exhibits a fair performance only when an in-scene pixel is used as target. On the contrary, ACE has a poor performance when the target signature of the material of interest is being detected.

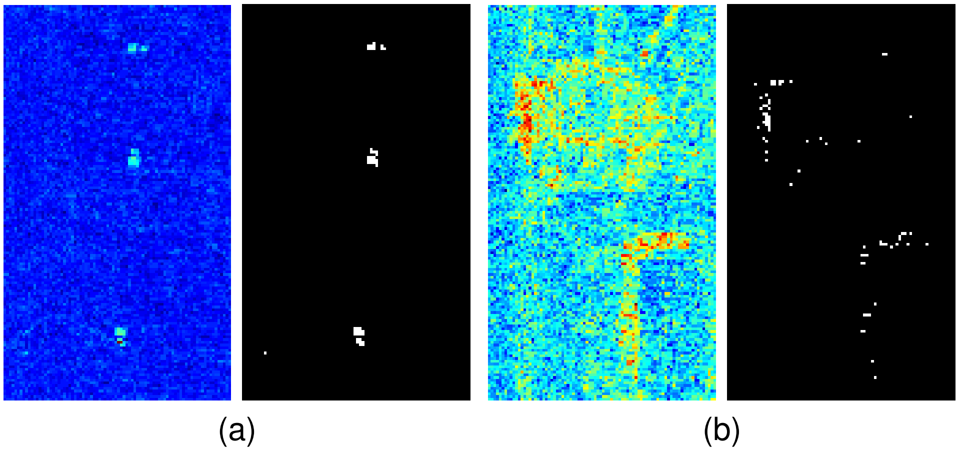


Fig 16 Results for ACE detector. Detection map image and its thresholded version. (a) In-scene pixel target. (b) Target Spectral Signature

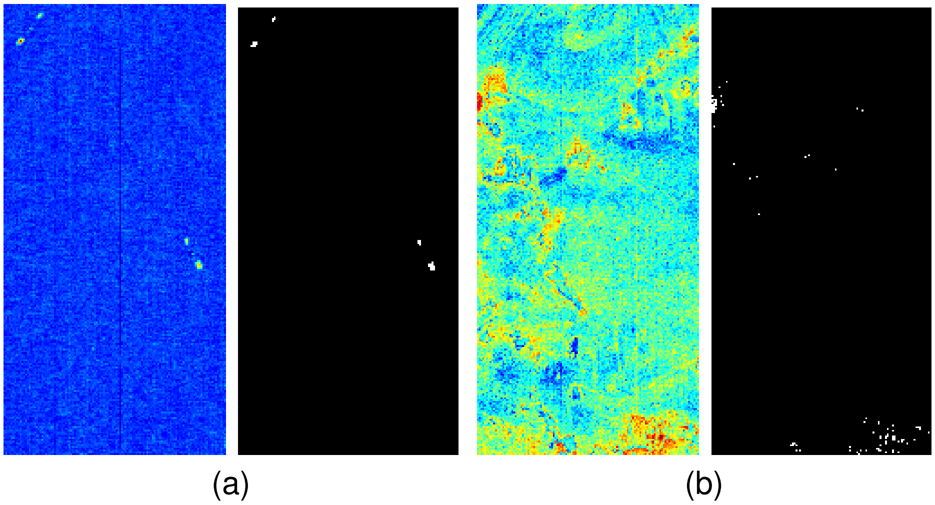


Fig 17 Results for ACE detector. Detection map image and its thresholded version. (a) In-scene pixel target. (b) Target Spectral Signature

4.3 Target Detection in Schroedinger Space - Quantitative Analysis

To evaluate the methodology proposed here from a quantitative point of view, we use the False Alarm rate (FAr) and the ROC curve. The FAr is given by the False Alarm score (FAs) divided by the total number of non target pixels. The FAs gives information about how many pixels are in the detection map greater than or equal to a detection value on a known target. The idea of using FAr instead of FAs is to have a scaled metric that facilitates a comparative evaluation. Given that in the computation of the FAs, the target value itself is considered, a perfect score would have a FAs of 1, indicating no false alarms and 1 target are present. Therefore, FAr will be scaled between 3.26×10^{-5} and 1 for SHARE 2012 image (the total number of pixels is 30750 and the number of true target pixels is 77) and 8.20×10^{-5} and 1 for SHARE 2010 (total number of pixels is 12264 and the number of true target pixels is 66). FAr $= 1$ corresponds to the worst performance (i.e., all pixels are detected). For our quantitative assessment, we define three different target pixels as reference target values. This will indicate detection performance relative to "easy", "moderate", and "difficult" targets to detect. In this way, we use the maximum (i.e. the easiest pixel), the median (i.e. an intermediate pixel) and the minimum (i.e. the most difficult pixel) of all the true target pixel values for each detection map. FAr were computed for all cases shown in the Sec. 4.2. Tables 1 and 2 show the FAr for the easy, the moderate and the hardest target pixel, in In-scene detection and target signature detection for SHARE 2010 and SHARE 2012. The best rates for each type of target value in each table are marked in bold.

SHARE 2010 results shown in Table 1 indicates all approaches based on SE algorithm have a perfect FAr for the in-scene and target signature detection, when the easy target pixel is considered as the reference target value. For the moderate target value, some of the approaches based on SE exhibits a better performance than the ACE detector, in both cases, In-scene detection and Target signature detection. However, for the worst target value, ACE detector has the best performance when the In-scene detection is considered, which imply that the ACE detector has a better performance than the all other SE-based approaches. This is reflected in the ROC curves shown in Fig. 18a, where the best ROC curve belongs to the ACE detector. On the contrary, for the target signature detection, the Adp$k$ method with ED has the best FAr (0.85) for the worst target value; and for the moderate target value, various SE-based approaches, $k = 4$ with ED and $k = 4$, $k = 20$ with SAD, have a perfect FAr. These behaviors are also consistent with the ROC curves shown in

Fig. 18b, where the curve belongs to the Adp$k$-SAD approach is the closest to the perfect detection corner (upper-left) and the worst ROC curve belongs to the ACE detector.

Table 1 False Alarm rates for SHARE 2010 and In-scene/Target signature detection. FAr are given for three different target values: easy-maximum of target pixels, mdrt-median of target pixels and wrst-minimum of target pixels.

In-Scene detection				
Similarity metric	**knn**	**FA rates**		
		easy	**mdrt**	**wrst**
Euclidean	**4**	$\mathbf{8.20 \times 10^{-5}}$	0.69	0.91
	20	$\mathbf{8.20 \times 10^{-5}}$	$\mathbf{8.20 \times 10^{-5}}$	0.91
	Adaptive	$\mathbf{8.20 \times 10^{-5}}$	0.61	0.92
Spectral Angle	**4**	$\mathbf{8.20 \times 10^{-5}}$	0.99	1.00
	20	$\mathbf{8.20 \times 10^{-5}}$	2.46×10^{-4}	0.97
	Adaptive	$\mathbf{8.20 \times 10^{-5}}$	2.46×10^{-4}	1.00
ACE method		$\mathbf{8.20 \times 10^{-5}}$	1.64×10^{-4}	$\mathbf{0.63}$
Target signature detection				
Similarity metric	**knn**	**FA rates**		
		easy	**mdrt**	**wrst**
Euclidean	**4**	$\mathbf{8.20 \times 10^{-5}}$	$\mathbf{8.20 \times 10^{-5}}$	0.90
	20	$\mathbf{8.20 \times 10^{-5}}$	0.29	1.00
	Adaptive	$\mathbf{8.20 \times 10^{-5}}$	0.16	$\mathbf{0.85}$
Spectral Angle	**4**	$\mathbf{8.20 \times 10^{-5}}$	0.99	1.00
	20	$\mathbf{8.20 \times 10^{-5}}$	$\mathbf{8.20 \times 10^{-5}}$	0.97
	Adaptive	$\mathbf{8.20 \times 10^{-5}}$	$\mathbf{8.20 \times 10^{-5}}$	0.98
ACE method		3.08×10^{-2}	0.45	0.98

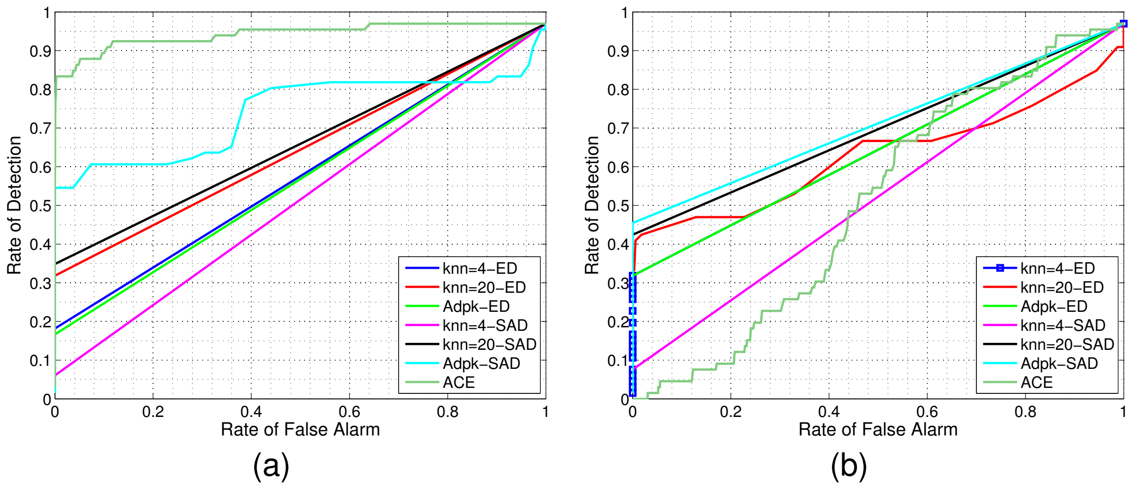


Fig 18 ROC curves for all approaches based on SE and ACE detector for SHARE 2010. (a) In-Scene pixel detection. (b) Target signature detection

These results have shown that although ACE detector outperforms all the SE-based detection approaches in the In-scene detection, ACE is defeated by all the SE-based approaches when the target signature detection is considered. Most of the approaches for the detection of the most difficult pixel, have very high FAr, indicating that this target pixel is practically undetectable.

Regarding **SHARE 2012 results**, Table 2 shows FAr for the In-scene detection and Target signature detection. For the In-scene detection, all approaches based on SE algorithm and ACE detector have a perfect FAr when the easiest target pixel is detected meaning that target pixel has the highest detection score in the image. For the intermediate target value, the ACE detector exhibits the best performance followed by the approach with SAD and $k = 20$. For the detection of the most difficult target value, there are two approaches with the best performance, these are the ACE detector and the $k = 4$-ED approach. The corresponding ROC curves are shown in Fig. 19a, where the ACE detector's ROC curve is the closer to the perfect curve, and for the SE-based approaches, the $k = 20$-SAD exhibits the second best performance. For the target signature detection, the rates in general are better than those for the In-scene detection. The rates for the intermediate target value are improved or at least are equal to the ones obtained in the In-scene detection, except for the ACE detector whose performance is decreased. For the most challenging target value, the $k = 20$-SAD approach achieves the best FAr, although the FAr are in general very high. These behaviors coincide with the ROC curves shown in Fig. 19b, where the SE-based approach $k = 20$-SAD has the ROC curve closer to the perfect detection curve.

Table 2 False Alarm rates for SHARE 2012 and In-scene/Target signature detection. FAr are given for three different target values: easy-maximum of target pixels, mdrt-median of target pixels and wrst-minimum of target pixels.

In-Scene detection				
Similarity metric	**knn**	**FA rates**		
		easy	**mdrt**	**wrst**
Euclidean	**4**	$\mathbf{3.26 \times 10^{-5}}$	0.72	$\mathbf{0.90}$
	20	$\mathbf{3.26 \times 10^{-5}}$	0.65	0.93
	Adaptive	$\mathbf{3.26 \times 10^{-5}}$	0.96	0.99
Spectral Angle	**4**	$\mathbf{3.26 \times 10^{-5}}$	0.75	0.96
	20	$\mathbf{3.26 \times 10^{-5}}$	0.12	0.95
	Adaptive	$\mathbf{3.26 \times 10^{-5}}$	0.92	1.00
ACE method		$\mathbf{3.26 \times 10^{-5}}$	$\mathbf{2.26 \times 10^{-4}}$	$\mathbf{0.90}$
Target signature detection				
Similarity metric	**knn**	**FA rates**		
		easy	**mdrt**	**wrst**
Euclidean	**4**	$\mathbf{3.26 \times 10^{-5}}$	0.72	0.90
	20	$\mathbf{3.26 \times 10^{-5}}$	0.52	0.94
	Adaptive	$\mathbf{3.26 \times 10^{-5}}$	0.96	1.00
Spectral Angle	**4**	$\mathbf{3.26 \times 10^{-5}}$	0.40	0.96
	20	$\mathbf{3.26 \times 10^{-5}}$	$\mathbf{3.26 \times 10^{-5}}$	$\mathbf{0.88}$
	Adaptive	$\mathbf{3.26 \times 10^{-5}}$	0.91	1.00
ACE method		4.2×10^{-3}	0.35	0.99

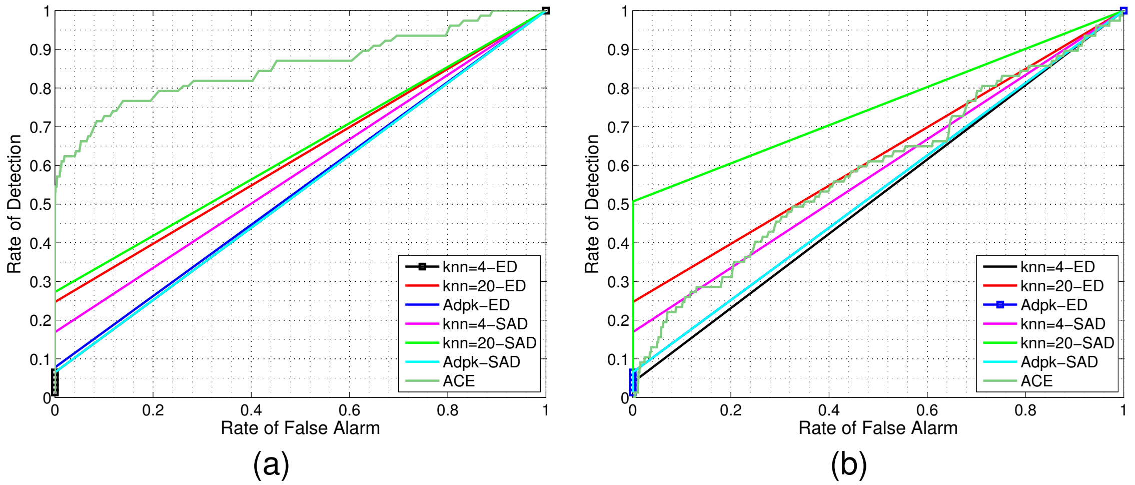


Fig 19 ROC curves for all approaches based on SE and ACE detector for SHARE 2012. (a) In-Scene pixel detection. (b) Target signature detection

4.4 Detection performance vs. Dimensionality in the Schroedinger Space

In order to analyze how the number of dimensions in the Schroedinger Space (SS) affects the detection performance, we use the FAr as a function of the number of Schroedinger eigenmaps used in the detection, i.e. a FAr value is computed when 1 SE is used in the detection, then a new FAr value is computed when 2 SEs are used in the detection, and so on until reach the maximum number of SEs (dimension in SS) estimated by Gram matrix method for each data set. The four plots in Fig. 20 correspond to the results for SHARE 2010 (in the first row), SHARE 2012 (in the second row) and for each case of detection, In-scene pixel detection (left column) and target signature detection (right column).

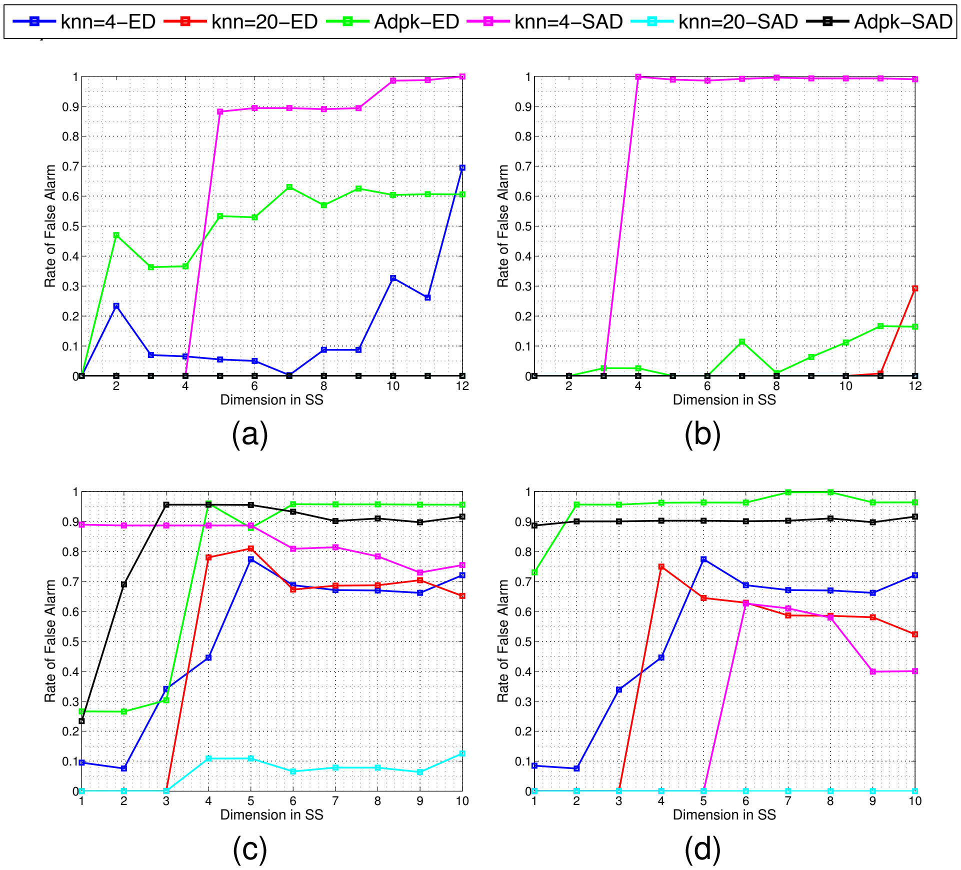


Fig 20 Plot of dimension in the Schroedinger Space vs. FA rates when an intermediate target value is detected. (a) SHARE 2010, In-Scene pixel detection. (b) SHARE 2010, Target signature detection. (c) SHARE 2012, In-Scene pixel detection. (d) SHARE 2012, Target signature detection.

Each plot in Fig. 20 includes the curves made up of the FAr values computed for each SE-based approach. The calculation of the FAr values is performed using the "the median" of all the true target pixel values as the target value to be detected. In each plot there are some FAr curves that are flat lines in zero along the x-axis, i.e. the detection performance is not affected by the number of SEs considered in each detection. However, for most of the approaches, especially in the detection of the target signature (second row), the FAr values start to increase when the number of SEs is also increased, which implies that the dimensionality in the Schroedinger space "poorly" influences the detection performance of detectors based on SE algorithm.

5 Summary and Conclusions

A methodology for spectral target detection based on graph theory is proposed here. This algorithm does not use a hypothesis testing approach but instead employs a data transformation with a *priori* knowledge of the target signature to identify likely targets. The methodology uses the SE algorithm to transform a data set in spectral domain to a Schroedinger Space, where the target class is not only separated from the background class but also its position in the Schroedinger space is predictable and close to the origin. This fact was used to define the detection operator that makes pixels around or close to zero have higher values than those pixels far from zero. Two frameworks were proposed: In-scene target pixel detection and Target signature detection. For each approach, two spectral metrics of similarity (ED and SAD) and different values for k were used. In addition, the diverse SE approaches were compared against the ACE detector that is one of the most widely used detectors; and two hyperspectral data sets, SHARE 2010 and SHARE 2012, were used for assessing the SE-based approaches from a qualitative and quantitative point of view .

Qualitatively speaking and for both data sets, the results show that in all of the considered

SE-based approaches at least one or two of the blue panels in the open field are detected with a few of false alarms. In general, SAD shows a better performance relative to the number of target panels that could be identified, especially for the target signature detection approaches, where even some of the target pixels in the shadow areas can be detected when SAD is used. In particular for SHARE 2012, there is only one case in which the seven blue panels are identified, this is when $k = 20$ and SAD are used. Comparing the SE algorithm with the ACE detector, the results support the better performance of SE in both cases, in-scene and target signature detection. Although, for the in-scene case, ACE can detect four panels, some of the SE approaches also find the same four panels (see Fig.12). For detection using the target signature, one SE approach, SAD and $k = 20$, identifies the seven blue panels, while ACE only shows false alarms. Regarding to SHARE 2010, the use of SAD in the target signature detection helps to identify one of the two blue panels located in shadows ($k = 20$ and Adp$k$), i.e. 6 out of 7 target panels are identified. The ACE detector in turn, exhibits a similar behavior than the ACE detector applied to the SHARE 2012 data set.

From the quantitative point of view, the ROC curves and False Alarm rate were used as a metric to assess the different approaches presented here. For the False Alarm rate, three different target pixels were used with the idea of evaluating the detection performance as a function of detection difficulty. According to the the FAr values, better results in both data sets were obtained when the target spectral signature was used. In general, SAD showed a better performance than ED. The best performances for all three cases (i.e. the easiest, the intermediate and the worst target pixel), were obtained when $k = 20$ and SAD were used. For the hardest target pixel, the FA rates are very high, almost 1 for most of the approaches. This indicates that this hard target pixel is practically undetectable. However, this does not mean that any detection at all could be achieved. In fact, we can still detect target pixels belonging to the target panels located in the open field, building's shadows (in SHARE 2010) and road (in SHARE 2012). Regarding the ROC curves, ACE detector exhibited the best curve in the In-scene pixel detection, but when the target signature detection was considered, the $k = 20$-SAD and the Adp$k$-SAD exhibited the best curves. In general, there is correlation between ROC curves and FAr values in all the approaches considered and the best detection was achieved when an in-scene pixel was used as target. Comparing the ACE with the SE approaches, the results showed that SE approaches (i.e. the ones that use SAD) outperform the ACE, particularly for the target signature detection approach. Therefore, the quantitative and qualitative analysis show that although there is not a consistent pattern of good detection for all the SE approaches shown here, the SE algorithm has potential to be applied to the problem of spectral detection.

In addition, a small analysis of how the dimensionality in the Schroedinger space affects the detection performance was also carried out. We found that the detection performance of many of the SE-based approaches are poorly impacted by the number of Schroedinger eigenmaps (SEs) used in the detection. This effect was also noticed in a previous work[34] where better visual and quantitative results for the same SHARE 2012 data set were obtained, but using in some approaches only a few number of SEs. Therefore, we can conclude that the detection performance is affected by the number of SEs used for computing the detection maps. Thus, there is not a consistency of the detection performance as a function of the dimension in the Schroedinger space. Consequently and as a future work, the use of SE in target detection has to continue being explored, with the goal of improving the detection performance even at high Schroedinger dimensions.

6 Acknowledgements

This project was partially supported by the Department of Energy Grant # PDP$12-059$.

References

1 D. Manolakis, C. Siracusa, and G. Shaw, “Hyperspectral subpixel target detection using the linear mixing model,” *Geoscience and Remote Sensing, IEEE Transactions on* **39**, 1392 – 1409 (2001).

2 Y. Ma and Y. Fu, *Manifold Learning Theory and Applications*, vol. 1st, CRC Press, Inc., Boca Raton, FL, USA (2011).

3 J. Khodr and R. Younes, “Dimensionality reduction on hyperspectral images: A comparative review based on artificial datas,” in *Image and Signal Processing (CISP), 2011 4th International Congress on*, **4**, 1875–1883 (2011).

4 R. Schowengerdt, *Remote Sensing Models and Methods for Image Processing*, Academic Press, Burlington, 3 ed. (2006).

5 L. J. P. Van der Maaten, E. O. Postma, and H. J. van den Herik, “Dimensionality reduction: A comparative review,” tech. rep., Tilburg University (2009).

6 M. Belkin and P. Niyogi, “Laplacian eigenmaps and spectral techniques for embedding and clustering.,” in *NIPS*, **14**, 585–591 (2001).

7 W. Czaja and M. Ehler, “Schroedinger eigenmaps for the analysis of biomedical data,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **35**, 1274–1280 (2013).

8 J. Benedetto, W. Czaja, J. Dobrosotskaya, T. Doster, K. Duke, and D. Gillis, “Semi-supervised learning of heterogeneous data in remote sensing imagery,” in *SPIE Defense+*

Security, *Proc. SPIE* **8401**, 840104–840104–12, International Society for Optics and Photonics (2012).

9 W. F. Basener, "Clutter and anomaly removal for enhanced target detection," in *SPIE Defense, Security, and Sensing*, 769525–769525, International Society for Optics and Photonics (2010).

10 C.-I. Chang, "Orthogonal subspace projection (osp) revisited: a comprehensive study and analysis," *Geoscience and Remote Sensing, IEEE Transactions on* **43**, 502–518 (2005).

11 D. Manolakis, D. Marden, and G. A. Shaw, "Hyperspectral image processing for automatic target detection applications," *Lincoln Laboratory Journal* **14**(1), 79–116 (2003).

12 L. L. Scharf and L. T. McWhorter, "Adaptive matched subspace detectors and adaptive coherence estimators," in *Signals, Systems and Computers, 1996. Conference Record of the Thirtieth Asilomar Conference on*, 1114–1117, IEEE (1996).

13 M. S. Halper, "Global, local, and stochastic background modeling for target detection in mixed pixels," in *SPIE Defense, Security, and Sensing*, 769527–769527, International Society for Optics and Photonics (2010).

14 B. Nadler, S. Lafon, R. R. Coifman, and I. G. Kevrekidis, "Diffusion Maps, Spectral Clustering and Eigenfunctions of Fokker-Planck operators," *ArXiv Mathematics e-prints* (2005).

15 J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science* **290**(5500), 2319–2323 (2000).

16 S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science* **290**(5500), 2323–2326 (2000).

17 M. Belkin and P. Niyogi, "Convergence of laplacian eigenmaps," in *NIPS*, (2006).

18 M. Belkin and P. Niyogi, “Semi-supervised learning on riemannian manifolds,” *Machine learning* **56**(1-3), 209–239 (2004).

19 X. Zhou and N. Srebro, “Error analysis of laplacian eigenmaps for semi-supervised learning,” in *International Conference on Artificial Intelligence and Statistics*, 901–908 (2011).

20 D. B. Gillis and J. H. Bowles, “Hyperspectral image segmentation using spatial-spectral graphs,” in *SPIE Defense, Security, and Sensing*, 83901Q–83901Q, International Society for Optics and Photonics (2012).

21 I. Tziakos, N. Laskaris, and S. Fotopoulos, “Multivariate image segmentation using laplacian eigenmaps,” in *Signal Processing Conference, 2004 12th European*, 945–948, IEEE (2004).

22 J. Shi and J. Malik, “Normalized cuts and image segmentation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **22**, 888–905 (2000).

23 L. Yang, “Data embedding techniques and applications,” in *Proceedings of the 2Nd International Workshop on Computer Vision Meets Databases*, *CVDB ’05*, 29–33, ACM, (New York, NY, USA) (2005).

24 X. Zou, Q. Zhu, and Y. Jin, “An adaptive neighborhood graph for lle algorithm without free-parameter,” *International Journal of Computer Applications* **16**(2), 20–23 (2011).

25 R. A. Mercovich, J. Albano, and D. Messinger, “Techniques for the graph representation of spectral imagery,” in *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2011 3rd Workshop on*, 1–4, IEEE (2011).

26 M. Belkin and P. Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation,” *Neural Computation* **15**, 1373–1396 (2003).

27 N. Cahill, D. Messinger, and W. Czaja, "Schroedinger eigenmaps with nondiagonal potentials for spatial-spectral clustering of hyperspectral imagery," *SPIE DSS Proceedings* (2014).

28 J. A. Albano and D. W. Messinger, "Euclidean commute time distance embedding and its application to spectral anomaly detection," in *SPIE Defense, Security, and Sensing*, 83902G–83902G, International Society for Optics and Photonics (2012).

29 A. K. Ziemann and D. W. Messinger, "An adaptive locally linear embedding manifold learning approach for hyperspectral target detection," in *SPIE Defense+ Security*, 94720O–94720O, International Society for Optics and Photonics (2015).

30 J. A. Bondy and U. S. R. Murty, *Graph theory*, vol. 6, Springer, London (2008).

31 L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in neural information processing systems*, 1601–1608 (2004).

32 U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing* **17**(4), 395–416 (2007).

33 J. A. Albano, D. W. Messinger, and S. R. Rotman, "Commute time distance transformation applied to spectral imagery and its utilization in material clustering," *Optical Engineering* **51**(7), 076202–1 (2012).

34 L. P. Dorado-Munoz and D. W. Messinger, "Schrodinger eigenmaps for spectral target detection," in *SPIE Defense+ Security*, 947211–947211, International Society for Optics and Photonics (2015).

35 D. Messinger, A. Ziemann, A. Schlamm, and B. Basener, "Spectral image complexity estimated through local convex hull volume," in *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2010 2nd Workshop on*, 1–4, IEEE (2010).

36 K. Canham, A. Schlamm, B. Basener, and D. Messinger, "High spatial resolution hyperspectral spatially adaptive endmember selection and spectral unmixing," in *SPIE Defense, Security, and Sensing*, 80481O–80481O, International Society for Optics and Photonics (2011).

37 J. A. Herweg, J. P. Kerekes, O. Weatherbee, D. Messinger, J. van Aardt, E. Ientilucci, Z. Ninkov, J. Faulring, N. Raqueño, and J. Meola, "Spectir hyperspectral airborne rochester experiment data collection campaign," in *SPIE Defense, Security, and Sensing*, 839028–839028, International Society for Optics and Photonics (2012).

38 A. Giannandrea, N. Raqueno, D. W. Messinger, J. Faulring, J. P. Kerekes, J. van Aardt, K. Canham, S. Hagstrom, E. Ontiveros, A. Gerace, *et al.*, "The share 2012 data campaign," in *SPIE Defense, Security, and Sensing*, 87430F–87430F, International Society for Optics and Photonics (2013).

39 C. Merkwirth, U. Parlitz, and W. Lauterborn, "Fast nearest-neighbor searching for nonlinear signal processing," *Physical Review E* **62**(2), 2089 (2000).

Leidy P. Dorado-Munoz is currently a PhD candidate at the Center for Imaging Science at the Rochester Institute of Technology, Rochester, New York. She received a BS degree in Physicis Engineering from Universidad del Cauca (Colombia) in 2006 and a MS degree in Electrical Engineering from University of Puerto Rico, Mayaguez in 2009. Currently, her research is focused on hyperspectral data exploitation by using graph-theory and manifold learning algorithms. Leidy is currently a member of SPIE and the IEEE Geoscience and Remote Sensing Society and she has served as a reviewer for the IEEE Transactions on Geoscience and Remote Sensing.

David W. Messinger received a Bachelors degree in Physics from Clarkson University and a Ph.D. in Physics from Rensselaer Polytechnic Institute. He is currently a Professor, the Xerox

Chair in Imaging Science, and Director of the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology where he previously was the Director of the Digital Imaging and Remote Sensing Laboratory. He is also an Associate Editor of the journal Optical Engineering. His personal research focuses on projects related to remotely sensed spectral image exploitation using physics-based approaches and advanced mathematical techniques with particular emphasis on the areas of large area search and target detection.

List of Figures

List of Tables