Spectral Target Detecting Using Schroedinger Eigenmaps

Leidy P. Dorado-Munoz
lpd5259@rit.edu

Follow this and additional works at: http://scholarworks.rit.edu/theses

Recommended Citation

This Dissertation is brought to you for free and open access by the Thesis/Dissertation Collections at RIT Scholar Works. It has been accepted for inclusion in Theses by an authorized administrator of RIT Scholar Works. For more information, please contact ritscholarworks@rit.edu.
SPECTRAL TARGET DETECTION USING SCHROEDINGER EIGENMAPS

by

Leidy P. Dorado-Munoz

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Imaging Science

Chester F. Carlson Center for Imaging Science
College of Science
Rochester Institute of Technology

July 7th, 2016
SPECTRAL TARGET DETECTION USING SCHRÖDINGER EIGENMAPS

by

Leidy P. Dorado-Munoz

B.S. Engineering Physics, Universidad del Cauca, 2006
M.S. Electrical Engineering, University of Puerto Rico-Mayagüez, 2009

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Chester F. Carlson Center for Imaging Science Rochester Institute of Technology

July 7th, 2016

Signature of the Author

Accepted by

Coordinator, Ph.D. Degree Program

Date
Ph.D. DEGREE DISSERTATION

The Ph.D. Degree Dissertation of Leidy P. Dorado-Munoz has been examined and approved by the dissertation committee as satisfactory for the dissertation required for the Ph.D. degree in Imaging Science

Dr. David Messinger, Dissertation Advisor

Dr. Nathan Cahill

Dr. Wojciech Czaja

Dr. Charles Bachman

Dr. Elizabeth Cherry

Date
Applications of optical remote sensing processes include environmental monitoring, military monitoring, meteorology, mapping, surveillance, etc. Many of these tasks include the detection of specific objects or materials, usually few or small, which are surrounded by other materials that clutter the scene and hide the relevant information. This target detection process has been boosted lately by the use of hyperspectral imagery (HSI) since its high spectral dimension provides more detailed spectral information that is desirable in data exploitation. Typical spectral target detectors rely on statistical or geometric models to characterize the spectral variability of the data. However, in many cases these parametric models do not fit well HSI data that impacts the detection performance.

On the other hand, non-linear transformation methods, mainly based on manifold learning algorithms, have shown a potential use in HSI transformation, dimensionality reduction and classification. In target detection, non-linear transformation algorithms are used as preprocessing techniques that transform the data to a more suitable lower dimensional space, where the statistical or geometric detectors are applied. One of these non-linear manifold methods is the Schrödinger Eigenmaps (SE) algorithm that has been introduced as a technique for semi-supervised classification. The core tool of the SE algorithm is the Schrödinger operator that includes a potential term that encodes prior information about the materials present in a scene, and enables the embedding to be steered in some convenient directions in order to cluster similar pixels together.

A completely novel target detection methodology based on SE algorithm is proposed for the first time in this thesis. The proposed methodology does not just
include the transformation of the data to a lower dimensional space but also in-
cludes the definition of a detector that capitalizes on the theory behind SE. The
fact that target pixels and those similar pixels are clustered in a predictable region
of the low-dimensional representation is used to define a decision rule that allows
one to identify target pixels over the rest of pixels in a given image. In addition,
a knowledge propagation scheme is used to combine spectral and spatial informa-
tion as a means to propagate the “potential constraints” to nearby points. The
propagation scheme is introduced to reinforce weak connections and improve the
separability between most of the target pixels and the background. Experiments
using different HSI data sets are carried out in order to test the proposed methodol-
y. The assessment is performed from a quantitative and qualitative point of view,
and by comparing the SE-based methodology against two other detection method-
ologies that use linear/non-linear algorithms as transformations and the well-known
Adaptive Coherence/Cosine Estimator (ACE) detector. Overall results show that
the SE-based detector outperforms the other two detection methodologies, which
indicates the usefulness of the SE transformation in spectral target detection prob-
lems.
Acknowledgements

I would like to express my most sincere gratitude to my advisor Dr. David Messinger, who has been an outstanding mentor whose advice have helped me to grow as a researcher and to accomplish this research project. I would also want to thanks my committee members: Dr. Nathan Cahill, Dr. Charles Bachman, Dr. Elizabeth Cherry, and Dr. Wojciech Czaja from University of Maryland, for their selfless advice, brilliant comments and suggestions in the preparation of this document.

A special thanks to the faculty in the Chester F. Carlson Center for their enthusiasm at teaching, for their exceptional and thoughtful lectures, and for their willingness to share their knowledge. Thanks to the Center and DIRS Lab staff, especially to Cindy Schultz, Sue Chan and Joyce French for always having their doors open for me, for helping me out with all kinds of paper work, and for always having a big welcoming smile.

I also want to thanks my family in the distance, my parents Olivia and Oscar, and my grandfather Cenén for their support, unconditional love, and for being always my role models. Thank you to my husband Harold for being the perfect companion, my best friend and my dearest love. I would not have been able to reach this stage without all of you.

Last but not least, thank you to my friends, the new ones, the old ones and the lifelong ones for letting me being part of your life, for letting me sharing moments of happiness, sadness and craziness, and for always being in one way or another a support for my life.
To my parents and family always
## Contents

1 Introduction .................................................. 17
   1.1 Motivation ............................................. 17
   1.2 Thesis aim and Contribution .......................... 19
   1.3 Thesis Outline ......................................... 20

2 The Target Detection Problem in Spectral Imagery .......... 22
   2.1 Optical Remote Sensing .................................. 22
      2.1.1 Hyperspectral Imaging .............................. 23
      2.1.2 Spectral variability modeling ....................... 26
   2.2 Target Detection using Spectral Information ............. 28
      2.2.1 Target Detection using Geometric Models .......... 29
      2.2.2 Target Detection using a Statistical Model ........ 31
      2.2.3 Assessing Target Detectors ......................... 32
   2.3 Summary .................................................. 35

3 Spectral Embedding Methods based on Manifold Learning ... 37
   3.1 Linear Embedding Methods .............................. 38
      3.1.1 Principal Component Transform ...................... 38
      3.1.2 Metric Multidimensional Scaling Transform .......... 39
   3.2 Graph Theory for Non-linear Spectral Embedding Methods 40
      3.2.1 Mathematical terms and definitions in Graph Theory 41
   3.3 Non-linear Embedding Methods ......................... 44
   3.4 Graph-based methods for HSI exploitation ................ 48
      3.4.1 The Topological Anomaly Detection (TAD) algorithm 48
   3.5 Summary .................................................. 49

4 Laplacian Eigenmaps for HSI exploitation ..................... 51
   4.1 LE algorithm ............................................ 52
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Diagram of the Electromagnetic spectrum with the three imaging systems covering ranges of wavelengths</td>
<td>24</td>
</tr>
<tr>
<td>2.2</td>
<td>Example of a hyperspectral cube with a typical spectral signal</td>
<td>25</td>
</tr>
<tr>
<td>2.3</td>
<td>Models for spectral variability of HSI: left, density model; middle, subspace model; right, linear mixing model.</td>
<td>26</td>
</tr>
<tr>
<td>2.4</td>
<td>Detection statistic histogram and definition of different regions involved in the detection process. The detection statistics are compared to a threshold ($\eta$) in order to decide whether the target is present or not present in the pixel. Figure adapted from [1].</td>
<td>29</td>
</tr>
<tr>
<td>2.5</td>
<td>ROC curve and its relationship with the theoretical background and target distributions. Figure adapted from [1].</td>
<td>34</td>
</tr>
<tr>
<td>2.6</td>
<td>Representation of a Detection Map and its relationship with the computation of FA score. When the easiest target pixel and the worst target pixel are chosen as target. For the last case, the false alarms are denoted with the letters FA.</td>
<td>35</td>
</tr>
<tr>
<td>3.1</td>
<td>Toy representation of different types of graphs. (a) Unweighted and Undirected Graph. (b) Weighted and Undirected Graph. (c) Unweighted and Directed Graph. (d) Weighted and Directed Graph.</td>
<td>42</td>
</tr>
<tr>
<td>3.2</td>
<td>2-D Synthetic data set. (a) Original data set is a sum of Gaussians with different means and equal covariances. (b) knn Graph with $k = 2$. (c) knn Graph with $k = 8$.</td>
<td>43</td>
</tr>
<tr>
<td>3.3</td>
<td>Classical linear PCT vs. non-linear ISOMAP. Original data set (left) in a three-dimensional space is embedded into two-dimensional space by using PCT (middle) and ISOMAP (right).</td>
<td>45</td>
</tr>
</tbody>
</table>
4.1 Toy data example in 3-d space. (a) Normal distributed dataset with three classes differentiated by color. (b) 2-d Principal Component Embedding, PC1 vs. PC2, (c) 2-d Laplacian Embedding, Φ₂ vs. Φ₃.

4.2 Small Hyperspectral scene (SHypS). (a) RGB rendering with tree classes framed in different colors. (b) Grass class representative spectrum. (c) Trees class representative spectrum. (d) Road class representative spectrum.

4.3 LE embedding for SHypS. (a) Laplacian Eigenvector 1. (b) Laplacian Eigenvector 2. (c) Laplacian Eigenvector 3. (d) 2D Scatterplot of LE1 vs. LE2. (e) 2D Scatterplot of LE2 vs. LE3. (f) 3D Scatterplot of LE1 vs. LE2 vs. LE3.

4.4 Flowchart of the target detection methodology based on LE.

4.5 LE in Target Detection. (a) RGB rendering of a Hyperspectral scene primarily covered by vegetation and with a few target panels. (b) Field-measure spectral signature of the blue target panel.

4.6 LE embedding for SpecTIR small scene. (a) Laplacian Eigenvector 1. (b) Laplacian Eigenvector 2. (c) Laplacian Eigenvector 3.

4.7 ACE detection output. (a) ACE detection image. (b) Histogram of ACE detection image with background pixels in blue color and target pixels in magenta color.

5.1 SE and LE applied to a toy data. (a) Original arc in 3D space. (b) Laplacian embedding. (c) Schrödinger embedding with α = 1. (d) Schrödinger embedding with α = 10. The Schrödinger embedding uses a barrier potential acting only in the middle point of the arc.

5.2 An example of a small set of points and its Schrödinger Eigenmaps with different barrier potentials. (a) RGB rendering of the scene. (b) 3D scatterplot displaying the pixels that are marked by blue color (background) and magenta squares (target) in the scene. (c) - (e) 3D Scatterplots of Schrödinger Eigenmaps for the marked pixels: (c) One target point is labeled as barrier potential. (d) Some target points are labeled. (e) All the target pixels are labeled.
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.3</td>
<td>(a) 3D scatterplot in the spectral domain of a small dataset with an added target spectral signature. The black point represents the target signature, magenta points are the target class and blue points are the background points. <strong>3D Scatterplot in the Schroedinger embedding:</strong> (b) with the target spectral signature position labeled as barrier potential, (c) the target signature and its spectral neighbors as barrier potentials.</td>
</tr>
<tr>
<td>5.4</td>
<td>Schroedinger Embedding for example in Figure 5.3c at different $\alpha$. 3D Scatterplots, blue color represents the background points, target points are in magenta color and the target spectrum is in black color. (a) $\alpha = 0.001$, (b) $\alpha = 1$, (c) $\alpha = 100$, (d) $\alpha = 1 \times 10^6$.</td>
</tr>
<tr>
<td>5.5</td>
<td>Knowledge propagation example. (a) RGB rendering of a small scene with true target pixels, spectral and spatial neighbors highlighted. Magnitude as function of the dimensionality of the Schroedinger embedding at various $\alpha/\beta$ values. Black lines are the background, magenta lines are targets and green line is the injected target spectrum. (b) $\alpha = 0.01$, (c) $\alpha = 0.11$, (d) $\alpha = 3.35$, (e) $\beta = 0.01$, (f) $\beta = 0.11$, and (g) $\beta = 3.35$.</td>
</tr>
<tr>
<td>5.6</td>
<td>Flowchart of the overall target detection methodology based on SE and Knowledge Propagation.</td>
</tr>
<tr>
<td>5.7</td>
<td>Flowchart of the adaptive $k$ estimation technique for the graph creation.</td>
</tr>
<tr>
<td>5.8</td>
<td>2-D Synthetic data set. (a) Original data set is a sum of Gaussians with different means and equal covariances. (b) knn Graph with $k = 2$. (c) knn Graph with $k = 8$. (d) knn Graph with Adaptive $k$.</td>
</tr>
<tr>
<td>5.9</td>
<td>Toy example for explaining the spectral and spatial Graphs creation. Target signatures are shown in gray color, actual target points in the data set are in magenta color, and the background points in blue color. (a) Data in spectral domain. (b) The spectral knn Graph built on the data. (c) Identification of target signature neighborhood, $sNN$, considering edge lengths $\leq 2$. (d) Over-connection of the target cluster – target signatures and their $sNN$ points.</td>
</tr>
<tr>
<td>5.10</td>
<td>The impact of using a single $\sigma$ or local $\sigma$ values. (a) Toy Data Set of two clusters with different local distribution. (b) Affinity between data points using a single $\sigma$ value. (c) Affinity between data points using local $\sigma$ values. The thickness of the edges represent the affinities between each point and its neighbors.</td>
</tr>
</tbody>
</table>
5.11 Decision boundary of the detector. (a) Toy Data Set of two normal distributed classes drawn in magenta and blue colors. (b) The Schroedinger embedding in a 2D space with the circle drawing the decision boundary. .................................................................................. 84

6.1 Self test data set. (a) HyMap Image including target F1. (b) Field measurement of F1 spectral signature. .................................................................................................................. 87

6.2 OSP Maps and histograms. (a) Approach using means. (b) Approach using EMs for 1st component. (c) Approach using EMs for 2nd component. .................................................................................................................. 88

6.3 OSP Maps and histograms. (a) Approach using EMs for 3rd component. (b) Approach using all EMs together. (c) Global Approach. .................................................................................................................. 89

6.4 ACE Maps and histograms. (a) Approach/all components together. (b)-(c) Approach/each component 2nd and 3rd components respectively. (d) Global Approach. .................................................................................................................. 90

6.5 SHARE 2010 showing blue and red panels locations. (a) RGB rendering with a 2% linear enhancement. (b) RGB rendering with a 255 linear enhancement. .................................................................................................................. 92

6.6 SHARE 2012 showing blue and red panels locations and target material spectra. (a) RGB rendering with a 2% linear enhancement. (b) Blue target spectra. (c) Red target spectra. .................................................................................................................. 93

6.7 MUUFL campus showing Pea green targets locations and target material spectra. (a) RGB rendering with a 2% linear enhancement showing in magenta the sub-pixel targets and in yellow the full pixel targets. (b) Pea green target spectral signature measured in the field and an in-scene pea green target spectrum. .................................................................................................................. 94

6.8 Flowchart for the experiment testing $\hat{\beta}$ value vs. the number of SE that can be used in the detection with a $\text{FAr} \leq 0.001$. .................................................................................................................. 96

6.9 SHARE 2010 detection results for blue material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE. .................................................................................................................. 98

6.10 SHARE 2010 detection results for red material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE. .................................................................................................................. 100
6.11 SHARE 2012 detection results for blue material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE.

6.12 SHARE 2012 detection results for red material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE.

6.13 MUUFL campus detection results for peagreen material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE.

6.14 SHARE 2012 with no blue material in the scene. (a) RGB rendering of the scene. (b) Diverse in-scene spectra and the blue material signature.

6.15 Detection maps and binary images with the 0.8% of the top pixels as threshold: (a) SE-SSKP and $k = 20$, (b) LE-ACE, (c) PCA-ACE.

6.16 RGB rendering of SHARE 2012 scene with magenta squares highlighting the locations of blue and red target panels, and blue arrows indicating the four blue panels injected in the target absent scene.

6.17 Target present vs. target absent hypothesis. (a) RGB rendering of a target absent scene and its response to the SE-SSKP, $k = 20$ detector represented by its histogram. (b) RGB rendering of a target present scene and its response to the SE-SSKP, $k = 20$ detector represented by the background histogram in blue and target stems in magenta.

6.18 Detection analysis considering the two hypothesis: target is present (right column)- target is absent (left column): (a) Background and target responses to the SE-SSKP, $k = 20$ detector represented by the background histogram in blue and by the target stems in magenta. (b) Binary images using a threshold of 1500 for both hypothesis. (c) Binary images using a threshold of 10500 for both hypothesis.

6.19 Number of false alarms vs. detection thresholds for the two hypothesis: target is present and target is absent.
List of Tables

6.1 Number of Schroedinger Eigenmaps at different $\hat{\beta}$ values . . . . . . . 96
6.2 SHARE 2010 results for blue and red material . . . . . . . . . . . . . 99
6.3 SHARE 2012 results for blue and red material . . . . . . . . . . . . . 104
6.4 MUUFL results for peagreen material . . . . . . . . . . . . . . . . . 107
6.5 Execution times for the detection techniques . . . . . . . . . . . . . . . 115
6.6 Size of the processed images . . . . . . . . . . . . . . . . . . . . . . . 116
# Nomenclature

## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACE</td>
<td>Adaptive Cosine/Coherence Estimator</td>
</tr>
<tr>
<td>Adpk</td>
<td>Adaptive $k$ nearest neighbor</td>
</tr>
<tr>
<td>AMD</td>
<td>Advanced Micro Devices</td>
</tr>
<tr>
<td>ASD</td>
<td>Adaptive Subspace Detector</td>
</tr>
<tr>
<td>ATRIA</td>
<td>Advanced TRiangle Inequality Algorithm</td>
</tr>
<tr>
<td>CEM</td>
<td>Constrained Energy Minimization</td>
</tr>
<tr>
<td>CTD</td>
<td>Commute Time Distance</td>
</tr>
<tr>
<td>cPMF</td>
<td>Constrained Positive Matrix Factorization</td>
</tr>
<tr>
<td>EMs</td>
<td>Endmembers</td>
</tr>
<tr>
<td>ENVI</td>
<td>ENvironment for Visualizing Images</td>
</tr>
<tr>
<td>FA</td>
<td>False Alarm</td>
</tr>
<tr>
<td>FAr</td>
<td>False Alarm rate</td>
</tr>
<tr>
<td>FAs</td>
<td>False Alarm score</td>
</tr>
<tr>
<td>FLAASH</td>
<td>Fast-Line-of-sight Atmospheric Analysis of Hypercubes</td>
</tr>
<tr>
<td>GLR</td>
<td>Generalized Likelihood Ratio</td>
</tr>
<tr>
<td>GLRT</td>
<td>Generalized Likelihood Ratio Test</td>
</tr>
<tr>
<td>HSI</td>
<td>HyperSpectral imaging</td>
</tr>
<tr>
<td>ISOMAP</td>
<td>ISOmetric Feature MAPping</td>
</tr>
<tr>
<td>JDQR</td>
<td>Jacobi-Davidson with QR decomposition</td>
</tr>
<tr>
<td>knn</td>
<td>$k$ nearest neighbors</td>
</tr>
<tr>
<td>LE</td>
<td>Laplacian Eigenmaps</td>
</tr>
<tr>
<td>LEs</td>
<td>Laplacian eigenvectors</td>
</tr>
<tr>
<td>LE-ACE</td>
<td>Laplacian eigenmaps with Adaptive Cosine/Coherence Estimator</td>
</tr>
<tr>
<td>LLE</td>
<td>Locally Linear Embedding</td>
</tr>
<tr>
<td>MATLAB</td>
<td>MATrix LABoratory</td>
</tr>
<tr>
<td>MaxD</td>
<td>Maximum Distance method for endmember selection</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MDS</td>
<td>Multidimensional Scaling</td>
</tr>
<tr>
<td>MODTRAN</td>
<td>Moderate resolution Atmospheric transmission radioactive transfer</td>
</tr>
<tr>
<td>MST</td>
<td>Minimum Spanning Tree</td>
</tr>
<tr>
<td>NIR</td>
<td>Near Infrared</td>
</tr>
<tr>
<td>OSP</td>
<td>Orthogonal Subspace Projection</td>
</tr>
<tr>
<td>PC</td>
<td>Principal Component</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>PCA-ACE</td>
<td>Principal Component Analysis with Adaptive Cosine/Coherence Estimator</td>
</tr>
<tr>
<td>PCT</td>
<td>Principal Component Transform</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Distribution Function</td>
</tr>
<tr>
<td>PMF</td>
<td>Positive Matrix Factorization</td>
</tr>
<tr>
<td>RGB</td>
<td>Red-Green-Blue regions in the electromagnetic spectrum</td>
</tr>
<tr>
<td>RIT</td>
<td>Rochester Institute of Technology</td>
</tr>
<tr>
<td>ROC</td>
<td>Receiver Operating Characteristic</td>
</tr>
<tr>
<td>RX</td>
<td>Reed-Xiaoli anomaly detection</td>
</tr>
<tr>
<td>SAD</td>
<td>Spectral angle distance</td>
</tr>
<tr>
<td>SE</td>
<td>Schroedinger Eigenmaps</td>
</tr>
<tr>
<td>SEs</td>
<td>Schroedinger eigenvectors</td>
</tr>
<tr>
<td>SE-SSKP</td>
<td>Schroedinger Eigenmaps with spatial-spectral knowledge propagation</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-noise-ratio</td>
</tr>
<tr>
<td>SSKP</td>
<td>Spatial-spectral knowledge propagation</td>
</tr>
<tr>
<td>SSSE</td>
<td>Spatial-spectral Schroedinger Eigenmaps</td>
</tr>
<tr>
<td>SWIR</td>
<td>Short Wavelength Infrared</td>
</tr>
<tr>
<td>TAD</td>
<td>Topological Anomaly Detector</td>
</tr>
<tr>
<td>VIS</td>
<td>Visible spectrum</td>
</tr>
</tbody>
</table>

### Parameters and Variables

- **A**: Adjacency matrix
- **a**: Abundances of background endmembers matrix
- **B**: Background endmembers matrix in geometric models
- **d**: Distance metric
- **D**: Degree matrix
- **D_{ss}**: Spatial-spectral degree matrix
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E$</td>
<td>Set of edges</td>
</tr>
<tr>
<td>$g$</td>
<td>Abundances of target endmembers matrix</td>
</tr>
<tr>
<td>$G$</td>
<td>Graph</td>
</tr>
<tr>
<td>$H$</td>
<td>Concatenated $B$ and $t$ matrix</td>
</tr>
<tr>
<td>$H_0$</td>
<td>Hypothesis for target absent in a test detection</td>
</tr>
<tr>
<td>$H_1$</td>
<td>Hypothesis for target present in a test detection</td>
</tr>
<tr>
<td>$I$</td>
<td>Identity matrix</td>
</tr>
<tr>
<td>$i$</td>
<td>index for pixels in a data matrix through rows</td>
</tr>
<tr>
<td>$j$</td>
<td>index for pixels in a data matrix through columns</td>
</tr>
<tr>
<td>$k$</td>
<td>number of neighbors considered in a nearest neighbor searching</td>
</tr>
<tr>
<td>$k_{\text{max}}$</td>
<td>Maximum number of nearest neighbors</td>
</tr>
<tr>
<td>$l$</td>
<td>index for endmembers in background or target endmembers matrices</td>
</tr>
<tr>
<td>$L$</td>
<td>Laplacian Operator</td>
</tr>
<tr>
<td>$M$</td>
<td>Total number of pixels in the data set</td>
</tr>
<tr>
<td>$m_b$</td>
<td>Total number of background pixels</td>
</tr>
<tr>
<td>$m_t$</td>
<td>Total number of target pixels</td>
</tr>
<tr>
<td>$m_{tt}$</td>
<td>Total number of target-objects</td>
</tr>
<tr>
<td>$N$</td>
<td>Total number of spectral bands of data set</td>
</tr>
<tr>
<td>$n_{\text{FN}}$</td>
<td>number of pixels misclassified</td>
</tr>
<tr>
<td>$n_{\text{TP}}$</td>
<td>number of pixels correctly classified as target</td>
</tr>
<tr>
<td>$n_{\text{TP}}^{*}$</td>
<td>Total number of pixels of correctly classified objects</td>
</tr>
<tr>
<td>$n_{\text{TOP}}$</td>
<td>number of objects correctly classified as target</td>
</tr>
<tr>
<td>$p(x</td>
<td>H_0)$</td>
</tr>
<tr>
<td>$p(x</td>
<td>H_1)$</td>
</tr>
<tr>
<td>$r_D$</td>
<td>rate of detection</td>
</tr>
<tr>
<td>$r_{FA}$</td>
<td>rate of False Alarms</td>
</tr>
<tr>
<td>$r_{FOA}$</td>
<td>Modified False Alarm rate</td>
</tr>
<tr>
<td>$r_{OD}$</td>
<td>rate of object-target detection</td>
</tr>
<tr>
<td>$S$</td>
<td>Schroedinger Operator</td>
</tr>
<tr>
<td>$s\text{NN}$</td>
<td>spectral Nearest Neighbors</td>
</tr>
<tr>
<td>$t$</td>
<td>Target endmember(s)/signature(s) matrix</td>
</tr>
<tr>
<td>$T_{SE}$</td>
<td>Detection operator based on SE</td>
</tr>
<tr>
<td>$V$</td>
<td>Set of vertices</td>
</tr>
<tr>
<td>$V$</td>
<td>Barrier potential matrix</td>
</tr>
<tr>
<td>$V_{ss}$</td>
<td>Modified potential matrix</td>
</tr>
<tr>
<td>$W$</td>
<td>Weighted adjacency matrix</td>
</tr>
<tr>
<td>$W_{PC}$</td>
<td>Eigenvector matrix for PC</td>
</tr>
<tr>
<td>$W_{sp}$</td>
<td>Spatial weighted matrix</td>
</tr>
</tbody>
</table>
**List of Tables**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W_{ss}$</td>
<td>Spatial-spectral weighted matrix</td>
</tr>
<tr>
<td>$x$</td>
<td>Pixel-vector sample in the spectral/original space</td>
</tr>
<tr>
<td>$y$</td>
<td>Pixel-vector sample after a transformation in a new space</td>
</tr>
</tbody>
</table>

**Greek letters and Special Characters**

- $\alpha$ Parameter of influencing potential matrix over Laplacian operator
- $\hat{\alpha}$ Scaled version for $\alpha$
- $\beta$ Parameter of influencing the modified potential matrix over Laplacian operator
- $\hat{\beta}$ Scaled version for $\beta$
- $\gamma_{ij}$ Spectral scaling factor between $x_i$ and $x_j$
- $\Gamma$ Background covariance matrix
- $\eta$ Threshold in detection hypothesis test
- $\eta_{it}$ Spectral scaling factor between $x_i$ and $x_t$
- $\Lambda$ Eigenvalue matrix of Schroedinger operator
- $\Lambda(x)$ Hypotheses test
- $\mu$ Background mean-vector
- $\Sigma$ Endmembers matrix for target class in statistics models
- $\Sigma a$ Target mean
- $\sigma$ Scaling parameter in weighted matrix computation
- $\Psi$ Eigenvector matrix of Schroedinger operator
- $\Phi$ Eigenvector matrix of Laplacian operator
- $\theta(x_i, x_j)$ Spectral angle distance between $x_i$ and $x_j$
- $\omega$ Gaussian noise
- $\omega(x_t, x_i)$ Spectral angle distance between $x_t$ and $x_i$
- $\ell$ Total dimension in the transformed lower/new space
- $\Xi$ Eigenvalue matrix of Schroedinger operator
Chapter 1

Introduction

A target detection methodology based on non-linear manifold learning techniques is proposed in this research work. The non-linear manifold learning method is the Schroedinger Eigenmaps algorithm that is used as a transformation but also as the theoretical framework for the definition of a detector based on a graph structure of the data. An introduction to the subject of interest including the motivation behind the study, its purpose and significance are provided here. In addition, a general overview of the document is presented at the end of this Chapter.

1.1 Motivation

Optical remote sensing is the area of research related to instrument-based techniques employed in the acquisition and measurement of objects properties on the Earth’s surface by means of a system that does not have physical contact with them. The output of this acquisition process is represented as images, hyperspectral imagery (HSI), with three dimensions, two spatial and one spectral covering a wide range of the electromagnetic spectrum. The optical sensors have a fine spectral resolution providing a spectrum for each pixel in the image, with the purpose to identify and distinguish materials. However, this also produces a high density data where redundant information could hide relevant features, and whose processing involves long processing time and high computational cost [2]. In order to reduce the processing costs and extract meaningful information from the data, it is desirable to transform it from its original spectral high dimensional space to a meaningful lower dimension. In this way, linear methods such as Principal Component Analysis (PCA) [3] have been widely applied. PCA finds the data representation in a lower dimension by describing as much of the data variability as possible, which in practical terms,
is performed by finding a linear uncorrelated basis that maximizes the covariance matrix of the data [4]. Once a more complex data is being processed, non-linear methods have shown better performance [5], given the inter-pixel distances in the feature space are preserved as best as possible in the new dimensional space. This process causes a natural clustering of the data while the dimensionality reduction is performed, what makes them appropriate to be used as clustering techniques. Many of these non-linear transformations are based on manifold learning techniques [6] that assume the original data lies on a low dimensional manifold embedded in a high dimensional space. One of these non-linear transformations is Schroedinger Eigenmaps (SE), which was introduced as a semi-supervised classification method based upon the well-known Laplacian Eigenmaps (LE) [7]. Both, LE and SE as non-linear manifold learning algorithms, involve the modeling of the local geometric structure of the data and the calculation of eigenvectors that serve as the lower dimensional representation of the data in the new space. The local structure modeling is performed by using an adjacency graph, i.e., a set of vertices (or nodes) and edges, where pixels from the data set are represented by the vertices and similar pixels are connected by edges. The eigendecomposition is performed on operators such as the Schroedinger operator [7,8] and the Laplacian [9] that are defined on the graph. The eigenvectors components are functions from vertices to $\mathbb{R}$ in the sense that nearby points are assigned “close by” values, and the eigenvalue of each eigenvector gives a measure of how “close by” are the values of nearby points. The difference between these two methods is the fact that the Schroedinger operator includes a potential term that steers the transformation in certain convenient directions that increase the separability among classes.

Besides classification, clustering or segmentation, target detection is also involved in the exploitation of spectral images where the identification and separation of a material is performed. The material has a specific and known spectral signature (i.e., target class) and is immersed in different materials with different spectra (i.e., background class). In general, the target detection methods are based on geometric or statistical models, which make assumptions about the behavior of the data, and define a hypothesis test framework in order to perform the decision whether a test pixel is a target pixel or not. However, one of their disadvantages is that in most of the cases the HSI data does not fit the models, which makes it convenient for the testing of non-linear embedding methods that have shown a satisfactory performance in the HSI processing. In this way, some of these graph-based methods have been used in target detection [10] and anomaly detection [11] (which is the identification of outliers, i.e., the spectral signature of the pixel of interest is unknown),
only as a means to transform the data to spaces where hidden features are exposed. Then, the detection is performed in those new spaces by using the classical detectors based on statistical or geometric models.

In this research work, the use of SE algorithm is proposed as a base for a target detection methodology that does not require assuming any statistical or geometric models but that boosts the separability between the class of interest and the other classes present in the image. This is performed by taking advantage of the Schroedinger Eigenmaps’ property of steering the embedding in some convenient direction. This steering is lead by the barrier potential term which is included in the definition of the Schroedinger operator. The barrier potential encodes the available “prior knowledge” about the target and because the embedding must preserve the local structure of the data set, these “potential constraints” are propagated to nearby points and all of them are pulled together toward a specific region in the new dimension-reduced representation. In addition, the fact that target pixels and their similar ones are clustered in a same region of the new lower representation is also taken into consideration to define a decision rule that allows one to identify target pixels over the rest of pixels in a given image.

1.2 Thesis aim and Contribution

This thesis is focused on the development of algorithms for the exploitation of HSI in optical remote sensing. The specific topic is target detection that is one of the most common applications of hyperspectral remote sensing data. The use of non-linear manifold algorithms has been previously introduced in spectral target detection. However, the non-linear methods are only used as pre-processing technique that transform the data to a new lower dimensional space, where relevant and important data features are better exposed. Then, in the lower dimensional space, a classical detector based on geometric or statistical models is applied to carry out the actual detection. On the contrary in this thesis, we are proposing for the first time a novel detection methodology completely based on a non-linear manifold algorithm. This novel method only uses a non-linear manifold algorithm for transforming the data and also for performing the actual detection. The specific non-linear manifold technique is the SE algorithm that does not only preserve the local structure of the data while the embedding is performed but also pushes together the target pixels and their nearby pixels in a specific and known region of the new lower representation.

Therefore, as the major contribution of this thesis, the use of the Schroedinger
Eigenmaps algorithm is introduced for the first time in spectral target detection. The use of this method in the detection methodology avoids the use of the classical detectors and the assumption of any statistical or geometric models that in many cases do not fit the HSI data. In addition, the designing of the method promotes three other contributions:

1. Spatial connectivity is integrated to the SE-based methodology by a “knowledge propagation” scheme. Spatial and spectral information are combined through a scheme that propagates the barrier potential constraints to spectral nearby points and their spatial neighbors, i.e., points connected via spectral and spatial adjacency graphs. This is performed in order to reinforce weak connections and improve the separability between most of the target pixels and the background pixels.

2. A detection rule that is not based on a hypothesis test is proposed. Typical target detectors are derived from competitive hypotheses that model the two scenarios: target is present or target is absent. On the contrary, the proposed detector takes advantage of the fact that the barrier potentials and its nearby pixels are pushed to zero in the dimension-reduced representation, so that, the detector is defined such that points close to zero will have higher values than those distant points.

3. An alternative way to estimate the \( k \) number involved in the \( \text{knn} \) searching for creating the spectral graph is proposed. This is an adaptive process in which the same \( k \) number is assigned to some pixels. The process is adaptive such that pixels in a high density region will have a high \( k \) value, and pixels in low density regions will have a smaller \( k \).

1.3 Thesis Outline

This thesis is distributed into seven Chapters that cover different topics related to the subject of interest of this research.

Chapter 2. An overview of basic concepts in the background subjects such as remote sensing and HSI are provided. The target detection problem is also established, including the classical methodologies proposed to solve this problem and the metrics used to assess the target detection methodologies.
Chapter 3. An introduction to the linear and non-linear spectral embedding methods based on manifold learning are described, as well as, the state-of-the-art and contributions from different research sources. The Chapter also presents an introduction to basic concepts of graph theory, which contributes to the understanding of the non-linear embedding methods, given there is a relationship between the manifold learning methods and spectral graph theory.

Chapter 4. Laplacian Eigenmaps algorithm is introduced together with the description of its characteristics for non-linear embedding. Its application to HSI processing tasks such as clustering and target detection are also presented including examples using synthetic data sets.

Chapter 5. Schroedinger Eigenmaps algorithm is described by using examples with synthetic data sets that show concepts and insights of the Schroedinger transformation, which can be taken into consideration for defining the target detection technique. The overall methodology based on the SE algorithm is also described in detail including the “knowledge propagation” scheme that is used to merge spatial information into the method.

Chapter 6. This Chapter includes experiments and results obtained after applying the proposed target methodology in different HSI data sets. Likewise, a preliminary methodology based on graph theory for target detection is also shown. The assessment includes qualitative and quantitative analysis for schemes detecting full-pixel and sub pixel targets.

Chapter 7. This is the final Chapter that presents a summary of all subjects and topics contained in this document. The insights, contributions and future work are also presented.
Chapter 2

The Target Detection Problem in Spectral Imagery

As it has been established in the introduction, the core subject of discussion in this dissertation is the detection and identification of a specific material immersed in several and diverse other materials. This Chapter as an opening for the discussion describes first the kind of data that is used in this work, as well as the acquisition system and the typical models used for representing their spectral variability. In the second part of the Chapter, the problem of target detection in the spectral domain is established, the classical methodologies proposed as possible solutions are explained, and the methods used for assessing the detection are also presented.

2.1 Optical Remote Sensing

The typical remote sensing system includes the use of sensors mounted on satellites or aircraft that collect data by detecting the energy reflected from the Earth’s surface. The type of sensor used in the acquisition defines what physical variables and techniques are involved in the process. There are generally two types of sensors and two types of remote sensing systems, passive and active. In this case, we are interested in the passive systems that unlike the active ones, do not need sensors with an internal energy source to illuminate the scene. In the passive sensors, the energy source is external, typically the sun, which stimulates the surface and the system collects the radiation reflected from it [2]. A typical optical system covers a wide range of the electromagnetic spectrum and includes a radiation source, an atmospheric path, the imaged surface and a system of sensors that captures the en-
energy that reaches it and converts it to digital form for further processing tasks. The propagation of the radiation starts from the sun (i.e. the source) and passes through the atmosphere, where due to absorption and scattering, its intensity and its spectral distribution are modified. Then, the energy that reaches the Earth’s surface interacts with different materials on the ground reflecting, transmitting and/or absorbing it. The reflected and/or emitted energy goes back up passing again through the atmosphere, where it is subjected to additional spectral absorptions and intensity modifications, to finally be captured by the sensors [1]. The acquired data is presented as an image that, in general, offers spatial, spectral, and temporal resolution.

The range of the electromagnetic spectrum covered by a typical optical sensing system includes regions such as the visible (0.4 – 0.7µm), the near (0.75 – 1.4µm) and short wavelength infrared (1.4 – 3.0µm) and depending on the number of used spectral channels, they can be classified as, Panchromatic imaging systems, Multispectral imaging systems and Hyperspectral imaging systems (HSI). The diagram in Figure 2.1 shows the regions in which the electromagnetic spectrum is commonly classified, as well as the three types of optical imaging systems. Panchromatic systems use sensors with a single channel detector sensitive to the radiation in a wide wavelength range. If this range coincides with the visible region, the physical quantity measured is the brightness of the imaged surface and the resulting image resembles a photograph in black and white. Meanwhile, the sensor in multispectral systems is a multichannel detector with a few spectral bands, where each channel is sensitive to the radiation in a narrow wavelength range. The resulting image is multilayered containing the brightness and the spectral information of the imaged surface. As it relates to HSI systems, the sensor has hundreds of detectors sensitive to narrower wavelength ranges, generating images in about hundreds of contiguous spectral bands. This more precise spectral information enables a better characterization of the different materials and objects present in the imaged surface [2]. This fact is taken into consideration by many processing techniques proposed for solving different image tasks such as classification or detection. HSI is explained in more detail below given it is the type of data that we are interested in.

2.1.1 Hyperspectral Imaging

HSI systems, also known as spectral imaging, collect and process information across the electromagnetic spectrum. The hyperspectral sensor system includes an optical system responsible to image the ground area into a number of contiguous squares
Chapter 2. The Target Detection Problem in Spectral Imagery

Figure 2.1: Diagram of the Electromagnetic spectrum with the three imaging systems covering ranges of wavelengths

(i.e., pixels) that together builds an image representing the spatial variations. At the same time, this system has a fine spectral resolution (i.e., hundreds of spectral detectors), which allows us to measure the radiation reflected by each pixel through the use of some sort of a dispersion element such as a prism or a grating that cover a wide range of electromagnetic spectrum, including different regions such as the VIS, NIR and SWIR (i.e., 0.4 – 2.5µm). Therefore, HSI will be a set of images, where each image represents the reflected radiation at one of those narrow bands. Figure 2.2 is an example of how the hyperspectral sensor system arranges spatially and spectrally the captured radiation. The HSI is then represented as a cube with two spatial dimensions and one spectral, where each pixel could be seen as a measurement of the integrated spectrum of the materials within that pixel. [2].
That measurement of radiance observed at certain wavelength is determined by the solar illumination and the reflectivity of the imaged materials at that wavelength, and it is affected by different effects that need to be considered when the analysis and processing of the information is performed. These effects include the angle of the sun and the viewing angle of the sensor, the radiance from the atmospheric scattering as well as the absorption by the atmosphere, shadowing, the spatial and spectral aberrations of the sensor, and the radiance interference produced by adjacent objects \[12\]. The atmospheric effects such as absorption and scattering are removed from the sensed imagery in order to retrieve the estimated surface reflectance. In this way, there has been proposed several atmospheric compensation algorithms based on radiative transfer modeling. One of these methods is the Fast Line-of-sight Atmospheric Analysis of Hypercubes (FLAASH) algorithm incorporated in the ENVI software package. FLAASH incorporates the correction for light scattered from adjacent pixels and the correction could be performed by choosing any of the standard model atmospheres and aerosol types that represent the scene \[13\]. These models are used by the Moderate resolution atmospheric transmission radioactive transfer code (MODTRAN), which is a software package that models the atmospheric propagation of electromagnetic radiation in a spectral range of \(0.2 – 100\mu m\) \[14\]. Once the atmospheric correction is performed, the hyperspectral image could be used in several processing and analysis tasks in different research areas such as agriculture, military, mineralogy, etc. The most common processing tasks include classification and segmentation, target and anomaly detection, and linear unmixing.
2.1.2 Spectral variability modeling

Material composition, location, and all factors previously mentioned that affect the captured radiation induce an inherent variability to the hyperspectral data that prevents the characterization of homogeneous surfaces by a unique spectral signature. There are three mathematical models that are typically used to help to characterize this spectral variability and model the data set. They are density model, subspace model and linear mixing model. Figure 2.3 shows a graphic description of the three models that are widely used in different algorithms, methods or techniques to perform different tasks such as classification and target detection.

![Figure 2.3: Models for spectral variability of HSI: left, density model; middle, subspace model; right, linear mixing model.](image)

**The density model** assumes that the probability that each observed data point $\mathbf{x}$ is generated is given by a probability mixture density,

$$ p(\mathbf{x}) = \sum_{\ell=1}^{\mathcal{L}} \pi_{\ell} p(\mathbf{x}|\ell), $$

where $p(\mathbf{x})$ is the probability of the spectral vector $\mathbf{x}$ occurring in the data set, $\mathcal{L}$ is the total number of classes in the data set with $\ell$ as index, $\pi_{\ell}$ is the prior probability of the $\ell^{th}$ class under the condition $\sum_{\ell=1}^{\mathcal{L}} \pi_{\ell} = 1$, and $p(\mathbf{x}|\ell)$ is the conditional probability of the data point $\mathbf{x}$ occurring given the $\ell^{th}$ class is observed. The conditional probabilities are usually chosen as a multivariate normal distribution, and the mean vectors and spectral covariance matrices must be estimated from the data set.
Chapter 2. The Target Detection Problem in Spectral Imagery

The subspace model assumes that the data set in $\mathbb{R}^N$ lies on a $\mathcal{L}$ dimensional subspace, with $\mathcal{L} < N$, so each data point $\mathbf{x}$ could be given by the linear combination of the basis vectors in the $\mathcal{L}$ space,

$$\mathbf{x} = \sum_{\ell=1}^{\mathcal{L}} a_\ell \mathbf{b}_\ell = \mathbf{B} \mathbf{a}, \quad (2.2)$$

where $\mathbf{b}_\ell$ is the $\ell$th basis vector with weight $a_\ell$. $\mathbf{B}$ and $\mathbf{a}$ are their matrix representation, where $\mathbf{a}$ is the vector of weights applied to the orthogonal basis vectors of the subspace, which form the columns of matrix $\mathbf{B}$.

The linear mixing model is also a model based on the geometric nature of the data set as the subspace model. It assumes that each pixel vector on the data set is a combination of pure materials called endmembers. The data set is also given by Equation (2.2), but here the basis in the new space has physical meaning. If we assume the pixels are linear mixtures of the pure materials then the subspace where the spectral data points lie is a convex hull generated by the endmembers. There are different algorithms to estimate the endmembers, some were mentioned before. The Positive Matrix Factorization (PMF) algorithm [15] is another, where the background spectra are constrained to be a positive linear combination of endmembers. Also the constrained PMF (cPMF) adds a sum-to-one constraint to the coefficients of the linear combination [16].

These three models have shown to be successful in different applications. In some cases and due to the randomness appearance of the spectra, the density model seems to better characterize the variability. However, the improved spatial and spectral resolution of the sensors have shown that the distribution of the data set is not as normal distributed as it seems to be, and the two models subspace and linear mixing make more sense to be applied. These two methods assume that the data set is generated by the linear combination of the basis vectors, but in cases where the materials are very close to each other, or are close to scattering surfaces, the spectra can be a non-linear combination of the pure materials, making it necessary to consider nonlinear mixing models to characterize the spectral variability[1]. Therefore, the modeling of the HSI spectral variability, and especially the modeling of the mixed pixels, are still some of the most challenging problems in HSI data exploitation.
2.2 Target Detection using Spectral Information

The detection of objects, movements or activities is very helpful in coastal and maritime surveillance, military monitoring, urban growth, etc. These problems could be addressed by target detection, which is one of the most important tasks in remote sensing and HSI data exploitation. The main goal of HSI target detection techniques is to find a known spectral signature in a hyperspectral image that contains many materials with different spectra. The known spectrum is called the target and the rest of the spectra of the different materials are known as background. In general, the target spectral signatures are obtained from target pixels in the image (i.e., in-scene target detection) or from library spectra or field/lab measurements (i.e., target spectral signature detection).

Mathematically speaking, the target detection problem can be defined in terms of a binary hypotheses test, where one of the two must be chosen. One of the hypotheses is related to the presence of the target ($H_1$) and the other one with the absence of it ($H_0$). The design of most of the detectors is based on the generalized likelihood ratio test (GLRT), because it minimizes the risk of an incorrect decision and maximizes the separation between target and background. The GLRT is written as

$$\Lambda(x) = \frac{p(x|H_0)}{p(x|H_1)},$$  \hspace{1cm} (2.3)$$

where $x$ is the observed pixel and $p(x|H_0)$ and $p(x|H_1)$ are the conditional probability density functions of $x$ under the two hypotheses. The target present hypothesis is selected as true when $\Lambda(x)$ exceeds certain threshold $\eta$.

Figure 2.4 shows the background and target distributions (in a simplified notional case) and how the threshold separates the two distributions. As was established before, HSI presents an inherent variability due to variations in the material surfaces and composition, variations of atmospheric conditions, sensor noise, surrounding materials and locations. Also, the pixels could cover just a single and pure material, or could be a mixture of pure materials. This is due to the finite spatial resolution of the sensor and the distribution of the materials on the ground [17]. Consequently, the different target detection algorithms that have been developed fall into one of the three models described in Section 2.1.2 or in some cases are a combination of them [14]. In general, the methods following the linear mixing and subspace model

28
Chapter 2. The Target Detection Problem in Spectral Imagery

2.2.1 Target Detection using Geometric Models

Geometric methods for target detection assume that most of the pixels in an image are a linear combination of pure materials called endmembers, so the entire spectral data set is formed by the set of the endmembers and the mixed pixels, whose proportionality is called the *abundance*. Since the mixing in the pixels is the result of a linear combination process, the data set forms a convex hull whose vertices are defined by the endmembers and the other mixed pixel vectors lay on the internal surface of the hull [14]. The binary hypotheses test is then written as

\[
H_0 : \quad x = Ba + \omega \quad \text{(target absent)} \\
H_1 : \quad x = gt + Ba + \omega \quad \text{(target present)}
\]  

(2.4)
Chapter 2. The Target Detection Problem in Spectral Imagery

where \( t \) is a column vector of the target spectral signature, \( B \) is a matrix of the background endmembers, \( g \) and \( a \) are the target and background abundances respectively, and \( \omega \) is a column vector associated with Gaussian noise. Different algorithms to choose between one of these two hypotheses have been developed. In general, these methods inhibit as much as possible the background interference and enhance the target contribution by projecting the data onto the subspace orthogonal to the background, where the residual energy of the target is high [1, 18]. However if the number of background basis vectors is overestimated, it is possible that the background includes target-signal components, which reduce the target residual energy in the orthogonal space. The most common algorithm based on the geometric perspective is the Orthogonal Subspace Projection (OSP) [18], and simultaneously, it has served as basis of other geometric detectors such as Adaptive Subspace Detector (ASD) [1].

OSP was designed to be effective in cases where the observed signal is a mixture of target and background. As a geometric detector, once background endmembers have been identified, OSP is formed basically by two steps: The first one attempts to minimize the effect of the background by projecting each vector \( x \) onto a subspace orthogonal to the background; and the second one tries to find anything target-like by applying a matched filter [14]. Thus, the OSP operator is a vector operating on the spectral vector \( x \) according to

\[
D_{OSP}(x) = t^T P_B^\perp x, \tag{2.5}
\]

where \( t \) is a spectral vector characterizing the target, \( B \) is a matrix made up of basis vectors spanning the background space, and \( P_B^\perp \) is the projection or nulling operator which minimizes the background contribution [14]. The nulling operator is given by, \( P_B^\perp = I - B B^\sharp \), where \( B^\sharp \) is the matrix pseudo-inverse given by \( B (B^T B)^{-1} B^T \). In addition, there is a normalized version of the OSP operator, which could be used to estimate the fractional abundance of the target in each pixel \( x \), and is given by

\[
D_{OSP}(x) = \frac{t^T P_B^\perp x}{t^T P_B^\perp t}. \tag{2.6}
\]

ASD was designed to have a constant-false-alarm rate. Therefore, ASD provides the same expected error levels along the data set by adapting to the local environment [1]. ASD is given by

\[
D_{ASD}(x) = \frac{x^T (P_B^\perp - P_H^\perp) x}{x^T P_H^\perp x}, \tag{2.7}
\]

30
Chapter 2. The Target Detection Problem in Spectral Imagery

where $H$ is a matrix that concatenates $B$ and $t$, and $P^\perp_H$ is its orthogonal projection operator. For this detector, a detection threshold can be determined theoretically or experimentally. Assuming a white noise Gaussian distribution, the theoretical threshold could be determined since ASD is $F$-distributed with $(1, N-1-P)$ degrees of freedom in the numerator and denominator, where $N$ is the number of spectral bands and $P$ is the number of basis vectors in $H$. Regarding the experimental threshold, it would be adjusted to achieve the desired false alarm rate by running the ASD in a large region where the number of targets are known [14].

2.2.2 Target Detection using a Statistical Model

These algorithms for target detection are based on the density model. These algorithms include the estimation of statistical parameters, such as the data mean-vector and covariance matrix, which help to characterize the image data described as a cluster of spectral vectors. In the detection algorithms, the data are modeled by a multivariate normal distribution with mean $\mu_0$ and covariance $\Gamma_0$ and in general, it is assumed that the background $B$ is a zero mean normal distribution that includes Gaussian noise, $B \sim N(0, \Gamma)$. The two detection hypotheses are given by

$$
H_0 : \quad x = B \quad \text{ (target absent)} \\
H_1 : \quad x = \Sigma a + B \quad \text{ (target present)}
$$

(2.8)

where $x \sim N(0, \Gamma)$ in $H_0$ and $x \sim N(\Sigma a, \Gamma)$ in $H_1$, whose mean for sub pixel detection is given in terms of $\Sigma$ and $a$ matrices that contain the available prior variability information about the target ($\Sigma$ is the endmembers matrix and $a$ is the abundances matrix of the target); and its covariance is equal to the background covariance [17]. The most common statistical detector is the spectral matched filter, which is a linear filter that correlates a known signal with an unknown signal to detect the presence of the first one, and it was initially proposed to optimize the signal-to-noise ratio (SNR) in radar/signal detection when the noise is the dominant source of variability. Variations to this algorithm have been proposed by the remote sensing community, where the variability of the signal is usually dominated by scene and spectral variations [14]. Two of the most used algorithms, Constrained Energy Minimization (CEM) [19] and Adaptive Coherence Estimator (ACE) [20] are described below.

CEM is basically the normalized spectral matched filter and its output yields an abundance that is scaled to unity when the pixel-value $x$ is equal to target-pixel
Chapter 2. The Target Detection Problem in Spectral Imagery

value $t$. The CEM detector is given by

$$D_{CEM}(x) = \frac{(t - \mu)^T \Gamma^{-1} (x - \mu)}{(t - \mu)^T \Gamma^{-1} (t - \mu)},$$

(2.9)

where $\mu$ and $\Gamma$ are the mean-vector and covariance matrix of the background generally estimated globally [14]. Like the OSP detector but from the other perspective, CEM tries to maximize the response of the target while suppressing the response of the unknown background.

ACE is a variation of generalized likelihood ratio (GLR), which works with estimations of background probability distribution functions (PDFs), because the background PDFs often cannot be characterized. The ACE detector could be expressed as

$$D_{ACE}(x) = \frac{(t - \mu)^T \Gamma^{-1} (x - \mu)}{\sqrt{(t - \mu)^T \Gamma^{-1} (t - \mu)} \sqrt{(x - \mu)^T \Gamma^{-1} (x - \mu)}}$$

(2.10)

with $\mu$ and $\Gamma$ as mean-vector and covariance matrix of the background, and $N$ is the spectral dimension of the data. Some implementations of the ACE detector consider that the terms in Equation (2.10) are squared, which could induce false alarms due to low negative detection scores that become large positive detection scores when they are squared [21].

2.2.3 Assessing Target Detectors

As any other task in HSI data exploitation, target detection needs to be assessed. The common way to assess detection includes the computation of rates of detection $r_D$ and false alarm $r_{FA}$, and the display of the relationship between these two rates in a Cartesian plane. This graph is known as Receiver Operating Characteristic (ROC) curve, which illustrates the performance of a binary classifier system when the thresholded is varied. In cases where the number of true target pixels is much smaller than the total number of pixels, ROC analysis is not meaningful, so it makes sense to have another assessment for the detector. In this way, the False Alarm (FA) score is a more meaningful tool to analyze the detection performance.

ROC curve

The ROC curve was initially proposed in signal detection theory. The curve represents the relationship between the rate of detecting correct positive pixels as target,
Chapter 2. The Target Detection Problem in Spectral Imagery

$r_D$, and the rate of detecting incorrect pixels as target, $r_{FA}$. These rates are given by $r_D = \frac{n_{TP}}{m_t}$ and $r_{FA} = \frac{n_{FN}}{m_b}$, where $n_{TP}$ is the number of pixels correctly classified as target, $m_t$ is the total number of target pixels in the scene, $n_{FN}$ is the number of pixels misclassified and $m_b$ is the total number of background pixels (i.e., the total number of pixels in the image ($m$) minus the total number of target pixels, which is given by the ground truth). Their calculation includes the use of prior knowledge about the position and size of the target materials in the scene. This information is commonly known as Ground Truth. The full ROC curve is obtained by varying the value of the threshold $\eta$ (see Figure 2.4).

Figure 2.5 shows a typical ROC curve and its relationship with the distributions of background and target classes, and with the threshold as well. As the threshold is varied, the detection probability and false alarm probability take different values, which generates a different point in the 2-D plane, making up the ROC curve. The worst case in the detection would be presented when the two distributions of background and target classes are identical, in terms of position and shape. In this case, the variation of the threshold value would increase the probabilities of detection and false alarm in the same quantity, causing a linear ROC curve with slope of 1.0. Contrary, the optimal ROC curve is obtained when the two distributions are far enough from each other, so the separability between them is enough to have the highest $r_D$ with a $r_{FA}$ of zero [17]. These two curves are also shown in Figure 2.5, as a dotted line and dotted curve.

**FA rate**

As it was said before, in cases where the total number of positive true target pixels is much smaller than the total number of pixels in the scene, it is required to use a different metric to assess the detection performance. The FA rate based on the FA score has been used in these cases. The FA score gives information about how many false alarms are obtained when a given target pixel is detected for a determined detection map. As with ROC curve computation, the ground truth is required to compute the FA score. There could be different approaches to compute the FA score depending on which target pixel is chosen as the target to be detected. For example, the target pixel with the smallest value in the detection map (i.e., the worst target pixel), or the pixel with the largest value (i.e., the easiest target pixel). Figure 2.6 shows a toy example of a detection map image with the target region marked in a green boundary, and the pixels chosen as target to be detected are denoted with a letter T. The figure on the left shows that the pixel target with the largest value is chosen as target, in this case no other pixel in the image has greater value than the
target, so there are not false alarms and the FA score would be perfect. The figure on the right shows that the pixel target with the smallest value is chosen as target, so in this case, there are some pixels with greater values than the target indicating there are false alarms and the FA score would have some value. FA rate in turn is just the ratio of FA score to the total number of pixels. This rate is introduced for comparative reasons, since the scaling of FA score makes easy the comparison between the performance of detection methods.

**Detection and False Alarm at Object Level**

As it has been described in this section, ROC curve and FA rate are pixel-level methods for analyzing the detection performance, i.e., detection and false alarm rates are calculated from a pixel-by-pixel counting. However, real and practical applications where large volumes of data are processed, do not require identifying every single pixel that conforms each object-target. The important objective is that the person analyzing the detection process can recognize the targets and exclude the non-target regions. Inspired by Theiler et al. work [22], we present a more practical evaluation for target detection where the assessment metric is based on the counting of target-objects. Therefore, the rate of object-target detection, $r_{OD}$, (analogous to the
Figure 2.6: Representation of a Detection Map and its relationship with the computation of FA score. When the easiest target pixel and the worst target pixel are chosen as target. For the last case, the false alarms are denoted with the letters FA rate of detection \( r_D \) is the rate of detecting correct positive objects as targets, \( r_{OD} = \frac{n_{TOP}}{m_{tt}} \), where \( n_{TOP} \) is the number of objects correctly classified as targets and \( m_{tt} \) is the total number of target-objects in the scene. One object-target is considered to be detected if at least one of its pixels is detected. Based on this \( r_{OD} \) rate, a modified false alarm rate, \( r_{FOA} \) can be computed. Its computation is achieved by counting the false-alarm pixels after removing from the detection map all pixels that are part of the whole object-target truly detected, \( r_{FOA} = \frac{m_{b} - n_{*TP}}{m_{b}} \), where \( n_{*TP} \) is the total number of pixels of the target object included in the \( n_{TOP} \) value.

### 2.3 Summary

This Chapter as the first part of the prelude to the detection method based on Schrodinger Eigenmaps contains a description of the kind of data of interest, i.e., HSI, and the exploitation task, i.e., target detection, that is treated in this work. The acquisition process of HSI data, including the physical variable that is sensed and the pre-processing involved in the calibration are presented, as well as the classical models to describe the HSI variability. The Chapter also describes the detection problem in spectral domain establishing the challenges and difficulties that must be taken into consideration by the classical target detection methods, which are primarily based on statistical and geometric models. At the end of the Chapter, the methods for evaluating the detection are also presented. Since the target detection method that we are proposing in this work, is based on non-linear manifold learning...
algorithms, (that do not make any statistical or geometric assumptions for modeling or representing the data) next Chapter will show the different data representation and transformation approaches that are commonly used in HSI exploitation, and that have shown potential use in processing tasks such as clustering or classification.
Chapter 3

Spectral Embedding Methods based on Manifold Learning

As it was mentioned in the previous Chapter, HSI is a highly dense dataset that can be difficult to interpret since it requires more than two or three dimensions to accurately represent. Moreover, high dimensional data typically contains redundant information that hides relevant and important features, which impacts the performance of different data-exploitation techniques. Therefore, it is convenient and appropriate to use methods for transforming the data set into spaces where the features are exposed and their interpretation is simpler. Most of the embedding methods are based on manifold learning algorithms that assume the data lies on a low-dimensional manifold and whose objective is to recover that manifold that it is embedded in the high dimensional space. These manifold learning techniques have been extensively used in statistical learning and data mining for achieving tasks such as classification or segmentation. The primary tools of the manifold embedding are the graph or adjacency matrix, which represent the local structure of the data; and the set of eigenvectors associated with the largest or smallest few eigenvalues of a specific matrix [6], which are used as basis vectors of the data-representation in the lower-dimensional space. The set of eigenvalues of a matrix/operator is known as the spectrum and given the importance of the eigenvalues/eigenvectors in the embedding process, the embedding methods are called as “spectral embedding methods”. This Chapter presents different spectral embedding methods based on manifold learning algorithms that are commonly used in HSI exploitation; propounds some differences among the algorithms and shows, how this methods besides the embedding are used as dimensionality reduction and clustering techniques.
Chapter 3. Spectral Embedding Methods based on Manifold Learning

3.1 Linear Embedding Methods

Spectral embedding methods are designed to recover linear or non-linear manifolds, although most of the data sets do not tend to lie on linear manifolds, i.e., lines, planes or hyperplanes depending on the number of dimensions involved. Therefore, there are two assumptions that we can use and that give support to this idea of recovering a linear manifold. One of the assumptions is that the data live close to a linear manifold. The other assumption is that the linear manifold is an approximation to a more complex non-linear manifold that is probably a better fit to the data. For both cases, it is taken that the intrinsic dimensionality of the linear manifold is smaller than the dimensionality of the data, implying that the linear manifold embedding is closely related to the linear dimensionality reduction problem. The linear dimensionality reduction problem is solved by deriving a set of low-dimensional projections of the original data that share some optimal properties [6].

In general, linear embedding methods are integral part of statistical toolboxes. The most common linear methods are Principal Component Transform (PCT) [3] and Multidimensional Scaling (MDS) [23]. PCA was originally proposed by Pearson in 1901 as an analogue system of axes that allows to represent points in a plane or in a higher dimensional space [24]. PCT has been widely used in computer vision helping in facial and object recognition, surveillance, person tracking and security [25, 26]. More recently, it has been also used in bioinformatics, where the first Principal Component (PC) of high dimensional genome data show specific classes of genes expressed by clusters of distinctive biological characteristics [27, 28]. MDS in turn, has been also used in bioinformatics, being useful in the display of a global representation of the protein-structure universe [29, 30], where points close to other points give information on the shape and function of proteins within the same family, which could be helpful in classification.

3.1.1 Principal Component Transform

PCT or Principal Component Analysis (PCA) has been the first projection method to be proposed for dimensionality reduction and it is the most popularly used. PCT was proposed as a technique for removing the spectral redundancy of multispectral data [31], since the analysis of all spectral bands results inefficient given the highly correlation between the different spectral variables that conform the multispectral image. Therefore, PCT can be also seen as a decorrelating transform. The linear embedding is then given by the set of orthogonal linear projections of $X$ that max-
imizes as much of the data variability as possible. Because the idea behind PCT is to maximize the variability of the data, the core of this technique is the covariance matrix that provides a notion of variance to multiple dimensions, and whose largest eigenvectors are the axis of the data projected onto the lower dimensional space. Assuming the multispectral/hyperspectral image $X = (x_1, x_2, ..., x_M)$ in $\mathbb{R}^N$ is organized as a matrix of size $[M \times N]$, the algorithm for finding the PCA embedding is described below.

1. **Covariance Matrix.** The Covariance matrix of the image $X$ is computed by the following equation:

$$
\Sigma = \frac{1}{N-1} \sum_{k=1}^{N} (x_k - \mu)(x_k - \mu)^T
$$

where $\mu$ is the mean vector given by $\mu = \frac{1}{N} \sum_{k=1}^{N} x_k$

2. **Eigenvector Matrix.** The eigenvectors and eigenvalues of Covariance matrix $\Sigma$ that satisfy the eigenproblem $\Sigma v = \lambda v$, are computed. Based on the eigenvalues, a matrix of eigenvectors $W_{PC}$ is formed, where only $\ell$ eigenvectors corresponding to the largest $\ell$ eigenvalues are kept, so $W_{PC}$ is a $[N \times \ell]$ matrix.

3. **Transformation.** Once the eigenvector matrix is computed, the dimensionality of the image is reduced by projecting the spectral bands onto a smaller subspace, where the $\ell$ eigenvectors are the axes of the new subspace. The projection is then given by,

$$
Y = XW_{PC}
$$

Notice that PCT has the property of redistributing the total image variance in the transformed data, so that the first PC contains the maximum possible variance of the original image, the second PC contains the maximum possible variance for any vector orthogonal to the first PC and so on. This enables to leave out the eigenvectors with the lowest eigenvalues that carry the least information about the variance of the data. Therefore, the $\ell$ number is set in order that the $\ell$ first eigenvalues capture some percentage of the total variance, where the percentage is usually set by an expert [2].

### 3.1.2 Metric Multidimensional Scaling Transform

Metric Multidimensional Scaling (MDS) is like PCA a classical approach of transformation and dimensionality reduction, and although they have different mathematics
involved, MDS yields a linear embedding as PCA. The dimensionality reduction in this case is performed as an attempt to preserve pairwise distances, i.e., the Euclidean distances between data points in the original space are much similar as possible to the Euclidean distances between the transformed data points. Mathematically, for an image $X$ in $\mathbb{R}^N$, the MDS embedding is given by

$$
\min_Y \sum_{i=1}^{M} \sum_{j=1}^{M} (d_{ij}^{(X)} - d_{ij}^{(Y)})^2,
$$

(3.3)

where $d_{ij}^{(X)} = \|x_i - x_j\|^2$ and $d_{ij}^{(Y)} = \|y_i - y_j\|^2$ and a distance matrix $D^{(X)}$ is made up by all the pairwise distance $d_{ij}$ between pixels in the original space. Since $D^{(X)}$ matrix can be rewritten as a kernel matrix of inner products $X^T X$, the minimization problem is reduced to

$$
\min_Y \sum_{i=1}^{M} \sum_{j=1}^{M} (x_i^T x_j - y_i^T y_j)^2.
$$

(3.4)

This minimization problem can be solved by finding the eigenvectors of the matrix $X^T X$ and the embedding in the lower space is then given by

$$
Y = \Lambda^{1/2} V_{xx}^T.
$$

(3.5)

where $\Lambda$ and $V_{xx}$ are the eigenvalues and eigenvectors of matrix $X^T X$. Note that MDS embedding can be seen as a map in a lower dimension that reconstructs the data structure as closely as possible, and although, MDS is a linear embedding, it is the core element of the non-linear embedding method, ISOMAP.

### 3.2 Graph Theory for Non-linear Spectral Embedding Methods

As it was previously mentioned, the non-linear manifold learning methods model the underlying low dimensional manifold from a group of data points by building an adjacency graph, then, before introducing some of the non-linear manifold learning methods, it is necessary to present mathematical definitions and concepts related to graph theory that are commonly used in the graph theory applications. The graph will be an approximation of the manifold and will represent as best as possible the data structure. The assumptions and inferences made about the graph structure can
be extrapolated to the manifold structure and simultaneously to the data structures. The first and the main step of nonlinear representation methods is the computation and analysis of operators or matrices that encode this structure information. Therefore, graph theory gives the mathematical foundations for these non-linear embedding methods, and some relevant definitions and mathematical terms are present below.

### 3.2.1 Mathematical terms and definitions in Graph Theory

Graph theory was introduced first as a method to represent the relationship between two variables directly related, and at the same time the relationship among all the variables in the data set. A graph could be seen as a diagram that shows the relation among variables by means of connections between them. For our specific case, the variables in the graph that are called vertices or nodes will represent each pixel-vector in a hyperspectral image, and the connections between a pair of vertices will represent their spectral similarity. Mathematically speaking, a graph \( G = (V, E) \) is a set of vertices, \( V = (v_1, ..., v_m) \) and edges, \( E = \{e_{ij}\} \). Thus, if there is some relation between nodes \( v_i \) and \( v_j \), there must be an edge \( e_{ij} \) in the graph \( G \) connecting them. There are also two kinds of graphs, weighted and unweighted. In an unweighted graph all the edges are equivalent (i.e., have value of 1), while each edge in a weighted graph has an associated value (i.e., weight \( w_{ij} \)), that in general is different from another edge’s weight. The weight is typically some measure of “similarity” between the nodes. There is also another categorization for the graphs, directed and undirected. 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 \( e_{ji} \in E \), so an undirected and weighted graph will have \( w_{ij} = w_{ji} \). Contrary to this, the order of the vertices in directed graphs is relevant, implying that the edges are not symmetric. Likewise, there is another term related to graphs called the connected component that is a subgraph in which any two nodes are connected to each other by paths or edges, and which is connected to no additional nodes in the graph. Figure 3.1 graphically shows the variables, definitions and relations for the different types of graph defined here.

In the representation of the graph, the nodes are represented by numbers 1, 2, ..., \( m \) and the edges are stored in an \( m \times m \) matrix \( A \), known as the adjacency matrix that represents the connectivity of the graph, i.e., \( A_{ij} = 1 \) if \( v_i \) is connected to \( v_j \), and \( A_{ij} = 0 \) otherwise. In general the adjacency matrix is used to represent the connectivity of unweighted graphs. There are different techniques used to build the
Figure 3.1: Toy representation of different types of graphs. (a) Unweighted and Undirected Graph. (b) Weighted and Undirected Graph. (c) Unweighted and Directed Graph. (d) Weighted and Directed Graph.

graph and compute the adjacency matrix. One of those is the $\epsilon-$ball graph, where each point is connected to points that lie within a ball of radius $\epsilon$. Another type of graph is the knn graph that is the most common and straightforward technique used in graph theory-based algorithms. In the knn graph, two vertices $v_i$ and $v_j$ are connected if one of them is among the $k$ nearest neighbors of the other one. In most of the cases, the number $k$ is arbitrarily and manually chosen. Although, choosing $k$ is very simple because it only takes integer values, this is its main disadvantage. Figure 3.2 is a graphic example of a toy data and graphs with different values of $k$. The data set was generated as a sum of points drawn from normal distributions with different means and the same covariance. The data set has two classes near-linearly separable and differentiated by color. Figures 5.8b and 5.8c show the graphs with $k = 2$ and $k = 8$ built on the toy data set. When $k$ is increased, the number of edges in the graph also increases, the choice of a right number for $k$ becomes then a trade-off problem, between a high number of $k$ that allows us to keep some important connections in the graph (that may resemble the local structure of the data), and the high computational cost and time that this implies. In this way, some variations of the knn graph have been introduced. A mutual k-nearest neighbor graph
connects the two vertices \( v_i \) and \( v_j \) if both \( v_i \) is among the \( k \) nearest neighbors of \( v_j \), and \( v_j \) is among the \( k \) neighbors of \( v_i \). This constraint of mutuality introduces a tendency to connect data points of similar density but leaving regions with different density unconnected. Therefore, variations to the mutual knn graph have been introduced by means of a minimum spanning tree algorithm (i.e., a minimum spanning tree (MST) is a subgraph with \( n - 1 \) nodes, of a graph with \( n \) nodes, that provides a path to every node and the sum of the weighted edges is a minimum), that connects the unconnected regions to other neighboring data points to finally have a connected graph \cite{33}.

For weighted graphs, the connectivity is represented by the weight matrix \( W \), whose entries are the weights of the associated edge, and 0 entries will represent that there is not an edge connecting the two vertices. In some cases, it is assumed that the vertices are not connected to themselves (i.e., there are no loops), so the elements on the main diagonal of \( A \) or \( W \) are all zero, (i.e., \( a_{i,i} = 0, w_{i,i} = 0 \)) \cite{32}. The weights represent how strong is the connection between two vertices, so if two vertices, \( v_i \) and \( v_j \) are very similar, they will have an edge connecting them with a high magnitude of weight, \( w_{ij} \). In similar fashion, if the two vertices are connected but their similarity level is low, the weight of the edge connecting those two will be very small. In general, the weight magnitude, \( w_{ij} \) is defined in terms of a pairwise similarity metric \( d_{ij} \) and different kernels that attempt to capture a more complex relationship between the vertices than a pairwise distance \cite{34}. The kernel most commonly used is the Gaussian kernel where the weight matrix is then given by
Chapter 3. Spectral Embedding Methods based on Manifold Learning

\[ W_{ij} = \begin{cases} 
 e^{-\frac{d_{ij}^2}{\sigma^2}} & \text{if } G_{ij} = 1 \\
 0 & \text{otherwise}
\end{cases} \]  

(3.6)

with \( \sigma \) as a scaling parameter that determines when two nodes are considered similar. The idea behind the use of the Gaussian kernel is to capture a more complex relationship between the connected nodes than \( d_{ij} \), that creates a better model of the local neighborhood relationship and better represents the local structure of the data set. In several techniques based on graphs, the parameter \( \sigma \) is manually set and it is based on a user criterion. In other cases the value is chosen from a range of values by trial and error, and even in those cases, the range of values for \( \sigma \) are manually set.

### 3.3 Non-linear Embedding Methods

It was previously established that linear embedding assumes that the data lies close to a linear manifold or that this linear manifold could be seen as an approximation to a more complex non-linear manifold. Thereby, in cases where the data is complex and the linearity assumptions do not work, the linear manifold embedding methods can not give a good representation for the data in the low dimensional space. It is necessary then, to consider that the data can lie on a non-linear manifold. Consider for example, the data set and its embeddings depicted in Figure 3.3. The “swissroll” in 3D space is embedded into a 2D space using classical linear and non-linear manifold techniques such as PCT and ISOMAP. The classical PCT fails to preserve the neighboring relationships between points of the swissroll while the ISOMAP properly “unfolds” the data preserving points of similar colors together. When points of different colors are next together, the embedding method does not respect neighboring relationships.

The non-linear embedding methods considered in this work are based on graph theory. For these non-linear embedding methods, the underlying manifold needs to be modeled from the data points, so it is necessary to approximate it by the construction of an adjacency graph on the data. The structure of this graph is then exploited by the non-linear manifold techniques, and because the graph is a proxy for the manifold, the findings about the graph structure can be extrapolated to the geometric structure of the manifold. Furthermore and as in linear embedding approaches, the non-linear techniques based on manifold learning assume that the
Chapter 3. Spectral Embedding Methods based on Manifold Learning

Figure 3.3: Classical linear PCT vs. non-linear ISOMAP. Original data set (left) in a three-dimensional space is embedded into two-dimensional space by using PCT (middle) and ISOMAP (right).

high dimensionality data lies on or near a manifold in a lower dimension. The goal then is to recover the representation of the unknown non-linear manifold that is embedded in the high dimension. The non-linear manifold learning algorithms most widely used and described here are Locally Linear Embedding (LLE) \[35\], Isometric Feature Mapping (ISOMAP) \[36\], and Laplacian Eigenmaps (LE) \[9\]. The data representation in the lower dimensional space for LLE and LE is given by the minimization of quadratic operators subject to normalization constraints. These non-linear manifold algorithms along with other algorithms such as Local Tangent Space Alignment (LTSA) \[37\], Hessian Eigenmaps (HLLE) \[38\], and Diffusion maps \[39\] share a similar scheme to find the embedding solution that is subject to some conditions on the underlying manifold \[40\].

Additionally, most of these methods seek to preserve the local structure of the data by considering only the relationship between each data point and a small number of its neighbors, which enables them to be used in classification.

ISOMAP

ISOMAP is one non-linear technique and it is the only one considered as a global technique because it uses the computation of geodesic distance (i.e., the shortest path) between all pairs of data points \[6\]. The ISOMAP embedding is given in the terms of the most faithfully preservation of pairwise distances between pixels. It assumes isometry and convexity, so that the Euclidean inter point distance in the low dimension is preserved. ISOMAP can be understood as a variation of MDS, in
which the Euclidean distances along the manifold are substituted for the estimates of geodesic distances. The ISOMAP Algorithm is described below:

1. First step is the graph creation by considering the $\epsilon$-ball technique, where the $\epsilon$ radius is measured from each data point; or by connecting each data point with its $k$ nearest neighbors.

2. The second step includes the computation of the edges’ weights which are the distances between the two points connected, and the estimation of the geodesic distances on the manifold by using the graph distances (i.e., shortest path distances between all pairs of data points in the graph).

3. The third step is the embedding where the geodesic distance on the manifold between two points is preserved as much as possible [36]. The actual representation in the lower dimension is then given by the $\ell$ eigenvectors corresponding to the most significant eigenvalues in the matrix $X^T X$, similar than MDS.

ISOMAP works well for intrinsically flat manifolds, i.e., surfaces that look like rolled up sheets, but it does not work if the manifold has holes within the roll because this is a violation to the convexity assumption. The preservation of the geodesic-distance in the ISOMAP embedding makes it suitable to be used as an algorithm for deriving manifold coordinates where a global optimal solution is guaranteed, but given the computational cost and memory requirements, ISOMAP is only practical when small datasets are processed. Therefore, some methodologies that deal with this computational issue have been proposed such as one technique proposed by Bachmann et al. (2005) [41], where the image is divided into small tiles and a manifold is derived for each single tile. Then, the manifolds are aligned and stitch together by using common tie-points to reintegrate the initial image. Along with this computational improvements, non-linear representations based on ISOMAP for larger HSI scenes can be derived, introducing more accurate representations that consider more details about the local curvature of datasets. However, the estimation of local curvatures induces a local intrinsic dimensionality, which is related to the size of the neighborhood (considered when the graph is built), and it can vary from region to region. These increase again the computational cost that in some cases is avoided by sampling the manifold, i.e., use a set of “landmarks”, on which the ISOMAP embedding is performed [42].
Locally Linear Embedding

Continuing with the non-linear embedding methods, another classical technique is Locally Linear Embedding (LLE). LLE performs the embedding in the lower dimensional space by preserving the local linear structure of nearby pixels in a dataset as faithful as possible. Contrary to ISOMAP, the LLE representation in the new space is given by the bottom (i.e., the ones corresponding to the smallest eigenvalues) eigenvectors of a sparse matrix \( [43] \). The algorithm includes also three steps:

1. First step is the searching for the nearest neighbors of each data pixel and the building of the graph by connecting each pixel to its neighbors.

2. Second step is the computation and assignation of weights to the edges of the graph, by describing each data point as a linear combination of its neighbors.

3. Third and final step is the finding of the embedding by using an eigenvector optimization where each point in the low dimension is still described with the same linear combination. Therefore, the actual solution is given by the \( \ell \) eigenvectors of the sparse matrix \((I - W)^T(I - W)\), with \( I \) as the identity matrix and \( W \) the matrix of weights.

Note that since LLE attempts to preserve the local neighborhood on the manifold, it is less susceptible to introduce false connections in the graph, which makes it more successful to embed non-convex manifolds [4]. Likewise, the fact that each data point is a linear combination of its neighbors makes LLE suitable to be used as a method for aligning manifold coordinates, especially in methodologies where the data is divided into tiles and the embedding is performed for each tile [41]. Another applications of LLE comprise supervised classification [44] and target detection of HSI data, where LLE is used as a method for transforming the data into a new space where relevant features are exposed. In target detection [10] specifically, a target manifold is artificially built an induced into the LLE technique for guiding the separation of the target from the background in the new space, and from there, Spectral Angle Mapper (SAM) algorithm is applied as target detector.

Laplacian Eigenmaps

Laplacian Eigenmaps (LE) is another non-linear manifold method to transform high dimensional data, and it will be described in more detail in the next Chapter. The core element of LE is the Laplacian matrix and the representation of the data in the
new space is given by its eigenvectors. Just as in LLE, LE is also a local approach, where the relationship between pairs of data points is established within a local neighborhood and the new representation may preserve as much as possible the local structure of the data set. As a nonlinear local transformation, it assumes that the data set \( X = \{x_1, x_2, \ldots, x_m\} \) in \( \mathbb{R}^N \) lies on a \( \ell \) manifold, where the dimension \( \ell \) is lower than \( N \). The main goal is to find the dimension-reduced representation \( Y = \{y_1, y_2, \ldots, y_m\} \) \[9\]. The LE algorithm as the other three previous non-linear embedding methods, also includes three basic steps:

1. The creation of the graph that generally is based on the searching for the \( k \) nearest neighbors. Although, another graph creation techniques can be considered such as the \( \epsilon \)-ball graphs.

2. The definition of Laplacian matrix that is an operator that exploits some particular structural characteristic about the graph and consequently about the data set.

3. The finding of the embedding in the new lower dimensional space that is given by the set of eigenvectors corresponding to the \( \ell \) smallest nontrivial eigenvalues of the Laplacian matrix.

### 3.4 Graph-based methods for HSI exploitation

In addition to the non-linear manifold methods, there are another methods that do not use a set of eigenvectors as the representation of the HSI data in a lower dimensional space, but still they are based on graph-theory. These methods have application in clustering analysis and in anomaly detection. The Topological Anomaly Detection (TAD) algorithm is one of these methods that have been proposed for anomaly detection and it will be explained below.

#### 3.4.1 The Topological Anomaly Detection (TAD) algorithm

Graph-based methods, as their name indicates, are methods that have as primary tool the representation of connectivity between data points. One of these methods is the Topological Anomaly Detection (TAD) algorithm that was initially proposed for anomaly detection \[45\]. Anomaly detection in some sense is similar to target detection, but the identification of the uncommon and unknown data pixel is performed without having information about it. TAD builds a graph model of the
background and computes an anomalousness scores for all of the pixels with respect to that background in order to find the anomalous pixels. This original implementation showed real improvements in HSI results compared against other traditional anomaly detectors, such as Reed-Xiaoli anomaly (RX) detector [46]. In this fashion, a graph-related analysis on TAD background model and its robustness is introduced in Ziemann et.al (2012). In that work, four different constructs of the graph: mutual k-nearest neighbor, sigma-local graph for two different values of sigma and the proximity graph are considered, as well as, different sample sizes on which the graph is built. For each kind of graph and sample size, TAD scores were computed and the results showed that the background model built in TAD is generally invariant to the sample size and graph structure [47], which speaks to the robustness of the graph-based model when is applied to HSI data.

Therefore, the use of TAD in other HSI exploitation problems has been explored. In this way, an application of TAD background model for classification was introduced by Fan et.al (2013) [48]. Another study of TAD applied to target detection was also introduced in [49], where the TAD background model is characterized in order to find geometric and statistical parameters that later is used in a classical detection framework. In the study, TAD estimates the background components and then that model is used as basis-vectors in geometric detectors, or as data for computing statistical parameters required in statistical detectors. This study was the first step in this research work, where the potential application of graph-based methods in target detection was explored. Some of the promised results will be shown next in the results Chapter.

3.5 Summary

This Chapter is divided into and presented in two important sections, the first one includes the classical linear transformation methods such as Principal Component Analysis and Multidimensional Scaling, as well as non-linear transformation methods based on manifold learning. The non-linear embedding methods include ISOMAP, Locally Linear Embedding and Laplacian Eigenmaps, since are the most representatives and are the base of methods for classification or even detection. The second part of the Chapter establishes some basic but relevant mathematical concepts in graph theory. These concepts are important since the non-linear embedding methods, which are the methods of interest, rely on a graph-based data representation to model the data local structure. As we continue presenting our proposed
method for Target detection, i.e., Schroedinger Eigenmaps, next Chapter brings in more details of Laplacian Eigenmaps since it is the base for Schroedinger Eigenmaps, and we would like to analyze at first hand its potential use in HSI exploitation.
Chapter 4

Laplacian Eigenmaps for HSI exploitation

It was shown in Chapter 3 that non-linear transformation methods do a better job representing the structure of the data set in a lower dimensional space. This fact makes them desirable to be applied to the processing of Hyperspectral imagery. In particular Laplacian Eigenmaps (LE) has shown its potential use in dimensionality reduction and classification [9]. As a non-linear embedding method, LE seeks to preserve the local structure of the data by considering only the relationship between each data point and a small number of its neighbors [50]. This local structure preservation leads to a natural clustering or segmentation as can be seen in Figure 4.1. In the figure, a normal distributed data toy made up by three different classes and in a 3D space is embedded into a 2D space by using the classical PCT and LE technique. LE embedding unlike PCT’s, shows a more organized distribution of the data points within each class, which induces a better separability among the three classes.

LE was firstly introduced by Belkin et.al. as a method for dimensionality reduction [50] and not long after it began to be used as a step in clustering techniques [9]. As a graph-based non-linear manifold embedding method, LE has the Laplace operator as a core element, which encodes the geometry of the data set through a graph-representation. This representation is an approximation of the underlaying manifold, the Graph Laplacian is in turn the approximation of the Laplace-Beltrami operator defined on the manifold [51, 52], and as in the other non-linear embedding methods, a set of eigenvectors of the Graph Laplacian is the representation of the data set in the lower dimensional space. This Chapter presents the classical and original LE algorithm that was initially proposed, as well as the substantiation behind
the use of the eigenproblem as a solution for the embedding. Later in this Chapter some specific applications of LE applied to HSI clustering and target detection are presented.

## 4.1 LE algorithm

The basic task of the LE algorithm is to generate a discrete approximation to a continuous map representing the geometry of a manifold. This algorithm is a simple and computationally efficient methodology that includes few local computations and a sparse eigenproblem. It is basically constituted by three steps: the creation of the graph, the definition of an operator that exploits some particular structural characteristic about the data set, and finding the embedding in the new dimensional space. Given a spectral image $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$ in $\mathbb{R}^N$ organized as a $[M \times N]$ matrix, its low-dimensional representation $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m\}$ is found by following the next steps [9]

1. **Graph construction.** Compute the adjacency matrix, $\mathbf{A}$ by any of the techniques described in Section 3.2.1. If the kNN graph is considered, the
searching for the $k$ nearest neighbors of each data pixel is performed with a $k$ heuristically set. Based on $A$, the weight matrix $W$ is computed by using the heat kernel given by the Equation (3.6), and the Euclidean distance as the pairwise similarity metric $d_{ij}$. The use of the heat kernel as a weight decay function is enabled by the key role that the Laplace-Beltrami operator has on differentiable functions and the heat equation.

2. **Computation of Laplace operator.** Once the weighted graph is built, the Graph Laplacian $L = D - W$ is computed, where $D$ is a diagonal weight matrix whose entries are column or row sums of $W$.

3. **Solving eigenvalue problem.** Find the mapping in the lower dimensional space by solving the eigenproblem $L\Phi = \Phi D\Lambda$. The optimal solution is the set of eigenvectors corresponding to the $\ell$ first smallest non-zero eigenvalues, where the trivial solutions, i.e., eigenvalue zero and its corresponding constant eigenvector is not considered.

### 4.1.1 The Optimal Embedding

The justification for the use of the eigenvectors of the Graph Laplacian as an optimal embedding that preserves local information in a certain sense [9] is shown below. It must be assumed that the graph is connected. Since the embedding given by the LE algorithm preserves local information in the sense that connected points are mapping to a line where they stay as close together as possible, a good choice is to minimize the objective function $\sum_{ij}(y_i - y_j)^2W_{ij}$, where $y = (y_1, y_2, ..., y_m)$ is the map in $\mathbb{R}$. Minimizing this objective function is an attempt to ensure that $y_i$ and $y_j$ are close if $x_i$ and $x_j$ are close as well. The objective function can be written as

$$\sum_{ij}(y_i - y_j)^2W_{ij} = \sum_i y_i^2D_{ii} + \sum_j y_j^2D_{jj} - 2\sum_{ij} y_iy_jW_{ij}$$

where $D_{ii} = \sum_j W_{ij}$. Since $W$ is symmetric and $L = D - W$, the expression (4.1) is equivalent to $2y^TLy$, with $y^TLy \geq 0$ indicating that $L$ is positive semidefinite. Then, the vector $y$ that minimizes the objective function is given by the minimum eigenvalue solution of $L\phi = \lambda D\phi$ subject to constraints that prevent the collapse of all vertices of $G$ onto any single point. Therefore, the minimization problem is written as

53
Chapter 4. Laplacian Eigenmaps for HSI exploitation

\[
\begin{align*}
\min_{y^T D y = 1, y^T D 1 = 0} y^T L y, \\
\end{align*}
\]

and the solution is now given by the eigenvectors corresponding to the smallest non-zero eigenvalues. The constraint \( y^T D 1 = 0 \) is added for removing a translation invariance in \( y \) [9]. Considering now the more general problem of embedding the graph into an \( \ell \) dimensional space, the embedding map is a matrix \( Y = \{y_1, y_2, \ldots, y_m\} \) in \( \mathbb{R}^{m \times \ell} \) with \( y_i \) yielding the embedding of the \( i \)th data point. The minimizing problem is then

\[
\begin{align*}
\min_{Y^T D Y = I} \sum_{ij} \|y_i - y_j\|^2 W_{ij} = \min_{Y^T D Y = I} \text{trace}(Y^T L Y),
\end{align*}
\]

where \( I \) is identity matrix and as for the one-dimensional embedding problem, the constraints in this case prevent the collapse onto a subspace of dimension less than \( \ell - 1 \). Similarly, the solution of the minimization problem is given by the matrix \( \Phi \) of eigenvectors corresponding to the smallest non-zero eigenvalues of the generalized problem \( L \Phi = D \Phi \Lambda \), with \( \Lambda \) as a diagonal matrix whose entries are the \( \ell \) eigenvalues.

### 4.2 LE for spectral clustering and segmentation

Since LE was proposed it has been shown to be of use in clustering and segmentation applications. However, LE was initially proposed as an embedding methodology just like other nonlinear manifold algorithms, where the primary goal is to find the low-dimensional representation of the original data, and the clustering applications have naturally emerged because the inherent preservation of the local structure [50]. Therefore, the first step in LE-based clustering methodologies is always the embedding step, where the reduced-dimension representation is given by the eigenvectors of the Laplacian. An example of how the Laplacian embedding looks like when LE is applied to real high dimensional data is shown below.

Figure 4.2 shows a real hyperspectral data set with 126 spectral bands covering the visible and infrared ranges of electromagnetic spectrum. The scene of interest is a small one of approximately 4000 pixels and it is part of a larger image. An RGB rendering is shown in Figure 4.2a and the spectral signatures of the principal materials that compound the scene are also shown in Figures 4.2b-4.2d. These materials are categorized as grass class, trees class and road class, and some of the pixels that constituted these classes are marked by squares with three different colors
in Figure 4.2a. The same colors are preserved in the spectral signatures plots (see Figures 4.2b-4.2d).

Figure 4.2: Small Hyperspectral scene (SHypS). (a) RGB rendering with tree classes framed in different colors. (b) Grass class representative spectrum. (c) Trees class representative spectrum. (d) Road class representative spectrum.

The LE embedding for the data shown in Figure 4.2 is presented in Figure 4.3. The first eigenvectors corresponding to the three smallest non-zero eigenvalues are shown in the first row. The second row shows the scatterplots between pairs of Laplacian eigenvectors (LEs), and for the three LEs in a 3D space. The images of LEs are displayed in grayscale where lighter areas have higher values in the Laplacian embedding. Therefore, we could associate the first LE to the grass class, the second LE to the trees class and the third LE to the road class. Likewise, the scatterplots in the bottom row, show the distribution of the pixels in the Laplacian embedding using specific axis. Pixels marked in colors in the Figure 4.2a are also represented with the same colors in the scatterplots. The first plot, which corresponds to the
two first LEs, displays a remarkable separation among the three classes. The second plot, i.e., second LE vs. third LE, contrary shows an inseparability between trees and grass. However, if the three first LEs are considered as it is shown in the 3D scatterplot, a defined separability among the classes is perceivable.

Figure 4.3: LE embedding for SHypS. (a) Laplacian Eigenvector 1. (b) Laplacian Eigenvector 2. (c) Laplacian Eigenvector 3. (d) 2D Scatterplot of LE1 vs. LE2. (e) 2D Scatterplot of LE2 vs. LE3. (f) 3D Scatterplot of LE1 vs. LE2 vs. LE3.

This sequence of LEs and scatterplots show the natural separation between classes, which is the potential use in clustering of materials present in high densely and complex data sets such as Hyperspectral imagery. That is why some more elaborate methodologies based on LE have been proposed for clustering, segmentation and classification. Clustering, the straightforward application of the Laplacian embedding refers to the grouping of pixels or materials in clusters such that the pixels/materials belonging to the same cluster are more similar (e.g., spectral similar) to each other than to pixels/materials in other clusters. The most known and widely used algorithm based on LE is Normalized Cuts that performs the clustering via graph portioning (this would be explained in detail later).
In addition, another extension of LE have been proposed for dealing with semi-supervised classification problems [54, 55] and segmentation [56]. Some of the improvements are related for example, with the use of different distance metrics to find the \( k \) nearest neighbors and build the graph [57], as well as, the use of adaptive algorithms for identifying the correct number of nearest neighbors [58, 59]; 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 [50].

Normalized Cuts Algorithm

One of the LE-based algorithm most widely used in segmentation problem is normalized cuts. Normalized cuts is a graph based segmentation method that addresses the segmentation as a graph partitioning problem. The method provides clusters such that edges connecting vertices within a cluster will have large weights, and the vertices connected by low-weights edges are disconnected and placed at different clusters. This partition of the data into clusters is given by the minimization of what is called “normalized cut”. A normalized cut is a measure of disassociation between subgroups of the graph given in terms of the sum of weights in each group [53]. Alternatively, the minimization problem can be redefined in terms of the Laplacian graph and the degree matrix, such that the minimization solution is obtained by solving the generalized eigenproblem, \( \mathbf{L}\Phi = \Phi\mathbf{D}\Lambda \), and the eigenvector with the second smallest eigenvalue is used to bipartition the scene after choosing an appropriate threshold. Finally and if it is necessary, the partition may continue iteratively [53].

4.2.1 Spatial-spectral fusion for classification and clustering

Graph-based methods and LE-based methods have proven to be useful in the analysis of hyperspectral imagery when the spectral information is explored. However, HSI also offers spatial information, such as objects size, objects shape or texture that can complement the spectral information in order to improve the discrimination and the identification of objects or materials. Consequently, joint spatial-spectral strategies can be used to generate more accurate results in HSI segmentation, clustering or classification. For example, in [60], a segmentation scheme based on an N-cut approach is proposed. The authors introduced the use of spatial connections for building the graph, but the weights were computed as a combination of the spa-
tial and spectral distances between the pixels involved in the connection. Then, the Laplace operator was used to recursively segment the image. The advantage introduce by this method comes from the use of a spatial graph instead of the usual spectral graph. This spatial graph introduces a more sparse weighted matrix, which facilitates the application of this method to large images. The analysis presented in the paper was performed across several HSI images and the results showed the utility of the spatial-spectral graph-based approach.

Another research works focused on LE-based dimensionality reduction or classification that handle spatial and spectral information, have introduced slightly differences in the way the spatial and spectral information is fused. In this way, Hou et al. [61] used a fused spectral-spatial distance to define the $k$ nearest neighbors of each pixel and the connections in the graph. The connections were defined such that an edge is created if the two pixels are mutually in the $k$ nearest neighbors of each other, and the weights were defined as binary weights. Likewise, Benedetto et al. [62] introduced a variable fused spatial-spectral distance to define the $k$ nearest neighbors and the weights in the graph. The variable distance uses a $\beta$ parameter ranging between 0 and 1 for tuning a Euclidean distance to be purely based on spatial information or spectral information, i.e., if $\beta = 0$ the distance is completely based on the spatial components and if $\beta = 1$ the distance will be completely based on the spectral components.

In other avenue of the spatial-spectral integration techniques, Schroedinger Eigenmaps (SE) will be briefly introduced since some spatial-spectral schemes have been introduced in SE-based classification techniques. The Laplace operator can be extended to an operator that includes a term called potential, which is capable of steering the embedding in some convenient directions that favors certain relations. This extension is the SE algorithm, which has been introduced as a semi-supervised classification scheme that enables the use of prior information encoded as labels in the potential term, and whose detailed description will be presented in the next Chapter. SE was introduced as a purely spectral technique for classification, however, Cahill et al. [63] introduced the Spatial-spectral Schroedinger Eigenmaps (SSSE) technique that uses spatial nondiagonal potentials in order to integrate spatial and spectral information in an unsupervised classification scheme based on a SE framework. The graph is completely built on the spectral information and the nondiagonal potential encodes the spatial relationships between pixels. This technique was applied to various HSI data sets and was compared against some prior LE-based algorithms, showing its superior/competitive performance. Furthermore, an extension of this
SE-based spatial-spectral technique was performed for cases where knowledge of the classes labels is partial. This partial knowledge impacts in the embedding and the classification performance since constraints imposed by the labels do not nicely propagate to the neighbors. The approach addressed this problem by replacing the original potential matrix, i.e., generated by the small set of labels, with a matrix that includes connectivity information given by the weighted adjacency matrix and the degree matrix. For testing the approach, a classification analysis with a comparative framework was carried out [64].

### 4.3 LE applied to target detection

Despite the good performance shown by LE in classification-related problems, it has not been as well tested in other HSI processing tasks such as target or anomaly detection. Target detection is our problem of interest, so it would be interesting to investigate the applicability of LE in target detection problems. One way to test this applicability has been shown in [11], where the “Commute Time Distance” (CTD) transform that is related the LE, is used as the basis for an anomaly detection technique. CTD is defined as the average length of a random walk between two vertices in a graph, when the walk starts from one of the vertices, transit to the other and come back to the starting one. This distance has the nice property of increasing when the number of paths or lengths between two vertices decreases, which is taken into advantage for the authors. The CTD transformation in turn, is closely related to the eigendecomposition of the pseudo inverse of the graph Laplace operator, allowing its use as a transformation that embeds the pixel-vertices of the graph into a Euclidean space with a separation between vertices equal to the square root of its average commute distance. This transformation has shown potential application also to clustering-related problems [33].

Another more straight strategy for applying LE to target or anomaly problems is to directly use LE as a non-linear transformation technique and then in the transformed space apply some detection algorithm such as classical detectors based on statistical models. This is the approach we are following here. First, the LE algorithm is applied to a hyperspectral image and to the spectral signature of the target material, and then a classical detector such as ACE detector is used to identify pixels that possibly contain the material of interest. A graphical description is shown in Figure [4.4](#) For testing this LE-based target detection methodology, we utilize a similar scene to the one used in the clustering example. In this case, the
scene is mainly covered by vegetation but also contains some target panels of red and blue colors. We are interested in the blue panels that are framed by magenta squares in Figure 4.5a and whose spectral signature measured in the field is shown in Figure 4.5b. The three images in Figures 4.6a-4.6c correspond to the three first LEs displayed in grayscale. The output of the target detection approach are shown in Figure 4.7.

![Flowchart of the target detection methodology based on LE.](image)

Figure 4.4: Flowchart of the target detection methodology based on LE.

![LE in Target Detection.](image)

Figure 4.5: LE in Target Detection. (a) RGB rendering of a Hyperspectral scene primarily covered by vegetation and with a few target panels. (b) Field-measure spectral signature of the blue target panel.

From visual inspection of the three LE images, it is impossible to distinguish the target panels. However, if we consider the image in Figure 4.7a that is the ACE detection map, only some pixels from one of the blue target panels are identified by ACE detector (this is the one in the right-bottom of the scene), the other two panels are completely missed. This is also evident from the histogram in Figure 4.7b.
where only a few of the magenta lines (corresponding to the target class) are well-separated from the background pixels depicted as blue lines. Although these results show once again the great potential of LE in HSI exploitation, there is still room for improvement. This is particularly true in cases where object-targets are harder to be identified (as the two target panels in shadows in Figure 4.5a) either because they are overshadowed or because the radiation from the surrounding objects interferes with the spectral signature of the target. It may then be desirable to somehow use prior knowledge about the target in order to improve the separation between the target and the background class.

Figure 4.6: LE embedding for SpecTIR small scene. (a) Laplacian Eigenvector 1. (b) Laplacian Eigenvector 2. (c) Laplacian Eigenvector 3.
Figure 4.7: ACE detection output. (a) ACE detection image. (b) Histogram of ACE detection image with background pixels in blue color and target pixels in magenta color.

4.4 Summary

This Chapter has presented the LE non-linear transformation methodology, and it has been divided into two parts. The first part is a description of the LE algorithm together with a mathematical analysis that gives support to its use as an optimal non-linear embedding. As second part of the Chapter, applications to HSI and some research work related to the use of LE in HSI processing are described. Real examples of LE applied to clustering and target detection of hyperspectral scenes are presented, summarizing findings and possible chances for improving, especially for target detection since it is the problem of interest for this research. This Chapter together with all the previous ones have described the information that gives support to the Schroedinger Eigenmaps, that is the non-linear manifold learning method we are proposing in this research work. The next chapter will explain the Schroedinger Eigenmap algorithm and the overall methodology for spectral target detection.
Chapter 5

Schroedinger Eigenmaps for Target Detection

Schroedinger Eigenmaps (SE) has been introduced as an extension or generalization of LE \[7\]. SE, like LE, is a non-linear manifold method whose core element is the Schroedinger operator just like the Graph Laplacian is for LE. The Schroedinger operator is built upon the Laplace operator and together with LE, SE has been used in classification problems due to the natural clustering that the embedding process induces in the new space [7]. This Chapter presents the Schroedinger Eigenmap algorithm that is the key element behind the detection methodology, as well as the overall methodology proposed for spectral target detection.

The main difference between LE and SE algorithms is that SE allows the use of labeled data with the purpose of influencing the diffusion process on graphs. The labeled points are pulled toward the origin in the dimension-reduced space according to the constraints imposed by the prior information, i.e., the labeled data. This prior information is encoded in the “potential matrix”, which is part of the definition of the Schroedinger operator, and it steers the embedding in directions where labeled data points and those similar are separated from the remainder pixels in the data set. This is an advantage in semi-supervised classification schemes [8], where SE has been shown to improve classification performance on imagery. Figure 5.1 shows a toy data example of 2D arc embedded into a 3D space, along with its Laplacian and Schroedinger embeddings. The Laplacian embedding perfectly rebuilds the 2D arc. For the Schroedinger embedding, the middle point, that is labeled in the potential matrix, is pulled toward the origin of the axis, and because the embedding must preserve the local geometry, the points surrounding the labeled data are also pushed
to the origin because they are connected to it in the graph.

Figure 5.1: SE and LE applied to a toy data. (a) Original arc in 3D space. (b) Laplacian embedding. (c) Schrodinger embedding with $\alpha = 1$. (d) Schrodinger embedding with $\alpha = 10$. The Schrodinger embedding uses a barrier potential acting only in the middle point of the arc.

In graphs where the spectral similarity between pairs of points is considered, not just the surrounding points of the labeled data are pulled toward zero, but also any other spatially distant data point, as long as it is a “neighbor” of the labeled data. This is very useful in hyperspectral images, where different pixels belonging to the same material are in general, spectrally similar but are not spatially connected. In Figure 5.1 the influence of the parameter $\alpha$ is also notable (see Equation (5.1)) in the embedding. As long as $\alpha$ is increased, the arc is folded around the middle point, pulling more points toward the origin. In general, this parameter is experimentally set so most of the labeled points and those similar are closer to zero.


5.1 Schroedinger Operator and SE Algorithm

As was established previously, SE is built upon LE, so SE can be seen as a generalization of LE. This is based on the fact that the Schroedinger operator is defined in terms of the Laplacian. Therefore, the Schroedinger operator is given by

\[ S = L + \alpha V \]  \hspace{1cm} (5.1)

where \( V \) is a potential term and \( \alpha \) is a scalar parameter which modifies the influence of the potential term over the Laplace operator. There are two types of potentials: Cluster Potential and Barrier Potential. The cluster potential is defined to have zeros everywhere except for the sub matrix defined by \( \iota = \{ i_1, ..., i_m^\prime \} \) that it is given by

\[
V[\iota, \iota] = \begin{bmatrix}
1 & -1 & & & \\
-1 & 2 & -1 & & \\
& & \ddots & \ddots & \ddots \\
& & & -1 & 2 & -1 \\
& & & & -1 & 1 \\
\end{bmatrix}.
\]  \hspace{1cm} (5.2)

The cluster potential is commonly used in applications related to classification or clustering. On the other hand, the barrier potential is defined to identify and better separate one class over the rest of classes present in the data set. The barrier potential is defined as a diagonal matrix \( V = \text{diag}(0, 1, 1, 0, 0, 1, ... 0) \) with ones and zeros along the main diagonal, where the ones indicate the position of the data points that belongs to the class of interest \( \mathbb{S} \), and work as constraints imposed on the set of points and those connected in the graph. We are specially interested in the use of the barrier potential because its natural and straight applicability to target detection, where the identification of one class over the rest of classes present in a specific scene is considered.

As a non-linear manifold learning method, SE’s main goal is to find a representation \( Y = (y_1, y_2, ..., y_m) \) of the original data set \( X = (x_1, x_2, ..., x_m) \), in a low-dimensional space. Because SE is a generalization of LE, the SE algorithm is based on the LE algorithm:

1. **Graph Creation**: a \( k \) nearest neighbor graph is built and its corresponding weight matrix is computed.
2. **Computation of Operators**: the Laplace operator and consecutively, the Schroedinger operator are computed.

3. **Eigendecomposition**: The mapping \( Y = (y_1, y_2, \ldots, y_m) \) as in LE, is computed by solving the eigenproblem \( S\Psi = D\Psi E \) and the set of eigenvectors, \( \Psi \), corresponding to the first \( \ell \) smallest eigenvalues in \( E \), conform the new dimension-reduced representation \([8]\), that are commonly known as “Schroedinger Eigenmaps”.

As in LE, the eigendecomposition of the Schroedinger operator is an optimal embedding for the problem of minimizing an objective function where local connectivity is preserved as best as possible. The minimization problem then is given in terms of the Schroedinger operator as:

\[
\min_{Y^T D Y = I} \text{trace}(Y^T (L + \alpha V) Y). \tag{5.3}
\]

Equation (5.3) could be rewritten for the barrier potential, \( V = \text{diag}(V_1, \ldots, V_m) \), as

\[
\min_{Y^T D Y = I} \frac{1}{2} \sum_{ij} \|y_i - y_j\|^2 W_{ij} + \alpha \sum_i V_i \|y_i\|^2. \tag{5.4}
\]

The first term in Equation (5.4) corresponds to the Laplace operator and the second term to the barrier potential. Analyzing the second term, if \( V_i \neq 0 \), where \( i \) corresponds to the spatial location of the labeled point in the data set, then \( \|y_i\|^2 \leq \frac{C}{\alpha V_i} \) (with \( C \) as a positive constant) and, as \( \alpha \to \infty \), \( y_i \to 0 \). This means that the labeled points are pulled toward 0 while the embedding is performed, and because the embedding must preserve the local structure of the data set, the neighbors of those labeled points (as connected via the graph \( G \)) are also pulled toward 0.

Figure 5.1 showed the effect of \( \alpha \) in the Schroedinger embedding when only one point was labeled as barrier potential. However, a question that remains unresolved is the impact of the barrier potential itself in the embedding, in other words, if the definition of the potential matrix \( V \) affects in different ways the embedding of a data set in the lower dimensional space. In this way, an analysis of how the Schroedinger embedding varies at different definition of the \( V \) matrix is carried out. Consider the example in Figure 5.2. An RGB rendering of a hyperspectral scene is shown in Figure 5.2a denoting pixels from different areas or materials marked with squares of two different colors. The magenta squares are used to denote the pixels from blue felt panels that are considered as the class of interest, i.e., the target class; and the
blue points denote pixels from other regions such as the open field of grass, regions of trees and its corresponding shadows, that conform the background class. Figure 5.2b in turn, is the scatterplot in a 3D spectral space of the pixels denoted in color squares. At this point, it is convenient to define $\alpha$ in terms of a parameter $\hat{\alpha}$ that is independent on the data set or the image being processed, $\alpha = \hat{\alpha} \frac{\text{Tr}(L)}{\text{Tr}(V)}$. Therefore, three different ways to define $V$ were considered while the $\hat{\alpha}$ parameter was set to the same arbitrary value 10. The Schroedinger embeddings for the small data set in Figure 5.2b are shown as 3D scatterplots in Figures 5.2c-5.2e. The first three Schroedinger Eigenmaps are used as axes. The definition of $V$ is given in terms of how many pixels from the target class (points in magenta) are labeled as barrier potentials. For the scatterplot in Figure 5.2c, one point was labeled in matrix $V$. For Figure 5.2d, three points were labeled as barrier potential in $V$, where each target point was picked from each target panel, and for the last case in Figure 5.2e, all the points from the target panels were labeled as barrier potentials in $V$.

Figure 5.2: An example of a small set of points and its Schroedinger Eigenmaps with different barrier potentials. (a) RGB rendering of the scene. (b) 3D scatterplot displaying the pixels that are marked by blue color (background) and magenta squares (target) in the scene. (c) - (e) 3D Scatterplots of Schroedinger Eigenmaps for the marked pixels: (c) One target point is labeled as barrier potential. (d) Some target points are labeled. (e) All the target pixels are labeled.
The three scatterplots show that as the number of target points labeled in the barrier potential matrix is increased (i.e., more 1 entries along the main diagonal of $V$ are considered), more target points are collapsed into the same point in the origin of the Schrödinger Eigenmaps, which coincides with our understanding of the Schrödinger embedding. However, it is noticeable that the separability between background and target points is affected, such that as more points are labeled in the barrier potential, the separability between those points and the remaining ones is decreased. This behavior can be explained in the sense of the connectivity constraints imposed via $G$, where the local structure of the data set is preserved as most as possible, i.e., the neighbors of labeled points are also pulled toward the origin.

This example has shown in some way the applicability of the SE algorithm and the use of barrier potential in target detection schemes. The problem that still remains in practical applications, is that the available information about the target class is a measurement of its spectrum, and the positions of pixels containing the target material are unknown. Therefore, it is convenient to combine this known target spectrum with the hyperspectral image that is processed. In this way, we propose to “inject” the target spectrum into the hyperspectral image, that has been previously sorted as a matrix of $[m \times n]$ where $m$ is the total number of pixels in the scene and $n$ is the number of spectral bands. Since the target spectrum is added at the end of the image matrix, the new size of the input matrix is $[m + 1 \times n]$, with the +1 position corresponding to the injected target spectrum. Figure 5.3a shows the scatterplot of the small data set (from the previous example) in the spectral domain, with the added target spectrum marked in black color. Figures 5.3b and 5.3c are the scatterplots in the 3D Schrödinger embedding, when only the position of the target spectrum (+1) is labeled in the $V$ matrix ($[m + 1 \times m + 1]$), and when the target spectrum position and the position of its spectral neighbors are labeled in $V$, respectively.
Figure 5.3: (a) 3D scatterplot in the spectral domain of a small dataset with an added target spectral signature. The black point represents the target signature, magenta points are the target class and blue points are the background points. 3D Scatterplot in the Schroedinger embedding: (b) with the target spectral signature position labeled as barrier potential, (c) the target signature and its spectral neighbors as barrier potentials.

According to the scatterplot in Figure 5.3b, only the target spectrum in black is exactly at the origin and the target points in magenta are pulled toward zero. Although the target points are pulled toward zero there is still a well-defined separation between the target spectrum in black and the target points in magenta. When the spectral neighbors of the target spectrum are also labeled in \( \mathbf{V} \) such as in Figure 5.3c, many more points of the target class are in and around zero; the well-defined separability between target points and the target spectrum, previously seen in Figure 5.3b, is reduced. However, there are still some target points very close to the background in blue color. These points are probably the pixels that share less similarity with the target spectrum such as sub-pixels or target pixels in shadow. Despite this, it is easy to realize that labeling multiple points in the barrier potential helps in pulling more points from the target class toward the origin of the Schroedinger embedding.
Another interesting thing to analyze is how the \( \hat{\alpha} \) parameter impacts the embedding when multiple points are labeled in \( V \). In this way, four different values of \( \hat{\alpha} \) are considered: 0.001, 1, 100, \( 1 \times 10^6 \). The idea is to have a wide range of values to see how the embedding behaves. Figure 5.4 shows the embeddings for the four cases, if the \( \hat{\alpha} \) value is small, \( \hat{\alpha} = 0.001 \), the target spectrum and many target points are pulled toward the origin but do not reach it yet. As \( \hat{\alpha} \) increases, the influence of \( V \) on the transformation is increased and the target spectrum and other similar points reach the zero and most target points are pulled toward that direction (see Figures 5.4b and 5.4c). This behavior stays until the alpha value is high enough to destabilize the system of equations involved in the eigenproblem, making that the solution does not converge (Figure 5.4d). Based on these toy examples and their Schroedinger embeddings, two things are concluded: one is that the \( \hat{\alpha} \) value has to be large enough to get as many points of interest as possible around zero, but not so large that the embedding may be compromised. The choice of \( \hat{\alpha} \) is then a trade-off problem. It also looks like the range of values for \( \hat{\alpha} \) that allows a stable behavior of
the embedding is wide enough to choose an appropriate value that greatly impacts the results. The second conclusion is that more than 1 point must be labeled in the barrier potential in order to get more points in and around zero of the Schroedinger embedding, especially in this case where the labeled point, whose location is known, is not itself part of the data set. In addition, it is convenient to try to combine more information into the barrier potential matrix that allows to pull more of the hard points (i.e., sub pixels, guard pixels that are spectrally close to the background) to the Schroedinger origin.

5.2 Knowledge Propagation for combining spatial and spectral contraints

Continuing with the analysis of the barrier potential, the Schroedinger embedding and its applicability to target detection problems. The aim proposed in this work, is to develop a methodology that boosts as best as possible the separability between most of the pixels belonging to the target class and those belonging to the background, as this would facilitate the identification and detection of most of the possible target pixels present in a given scene. In this way and considering the behavior of the Schroedinger embedding shown in the previous example, it is desirable to consider the use of another type of information that improves further the separability between difficult target pixels and the background. In a similar area of discussion, Cahill et.al. has shown the impact of the use of cluster potentials generated from small sets of labeled points in dimensionality reduction and classification. It was shown that despite the similarity of the labeled pixels in the low-dimensional space, their nearby pixels in the low dimensional space are not neighbors in the high dimension, i.e., the constraints represented by the labeled pixels in the cluster matrix, are not adequately propagated to the high-dimensional-neighbors. This propagation impacts in a “brittle” manner the dimensionality reduction and also may affect the classification performance. Cahill, et al. also show a way to improve the classification performance by propagating the constraints on the labeled pixels to nearby pixels by replacing the cluster potential by a matrix that includes more information about the connectivity of the labeled pixels. This scheme is called Knowledge Propagation and is based on the spectral clustering technique proposed by Yu and Shi.

Returning to the target detection problem, the barrier potentials are generated from the target spectral signature and its spectral neighbors. This barrier potential generation makes the $V$ matrix sparse, making it difficult that points near the
barrier potentials in the high-dimensional space, are identified with each other and with the potentials in the low-dimensional space. On these grounds, the knowledge propagation scheme appears to be suitable for this problem. In this way and inspired in Cahill’s work, the barrier potential matrix $V$, defined as a diagonal matrix with non-zero entries at the position of labels, i.e., barrier potential constraints, is modified to include spectral and spatial connectivity information of those labeled pixels in $V$. The combination of the spatial and spectral information into the “knowledge propagation” scheme is inspired by the examples shown in Figures 5.3 and 5.4 where the embeddings expose that spectral connectivity alone is not enough to define a significant separability between target and background classes, especially, for sub-pixel targets or target pixels in shadow that share less spectral similarity with the true full target pixels, but that spatially can be nearby. Therefore, the “knowledge propagation” matrix for spatial and spectral constraints is given by

$$V_{ss} = W_{ss}D_{ss}^{-1}VD_{ss}^{-1}W_{ss},$$  \hspace{1cm} (5.5)$$

where $W_{ss}$ and $D_{ss}$ are the spatial-spectral weighted adjacency matrix and its corresponding degree matrix. The use of spatial connectivity encourages points that are spatially close to the barrier potential points in the high-dimensional space to be also close together in the low-dimensional space, so that the constraints defined on individual points are propagated to spatial and spectral nearby points. This $W_{ss}$ matrix combines the spatial and spectral weights as a sum of both weighted matrices,

$$W_{ss} = W + W_{sp},$$  \hspace{1cm} (5.6)$$

where $W$ is the spectral weighted matrix computed for the total number of pixels in the scene, and $W_{sp}$ is the spatial weighted matrix defined on the entire image but with non-zero entries only at the position of the spatial neighbors of each barrier potential point (i.e., the spectral neighbors of the injected target signature). In addition, it is appropriate to “balance” the spatial weights by spectral and spatial similarities. This will strengthen or weaken the edges in the graph depending on the surrounded region of the barrier pixel that is considered,

$$W_{sp} = \gamma_{ij}\eta_{it} \times \begin{cases} e^{-\frac{d_{sp}(i,j)^2}{\sigma_{sp}^2}} & \text{if } j \in N_k(x_i) \\ 0 & \text{otherwise} \end{cases},$$  \hspace{1cm} (5.7)$$

with $N_k(x_i) = \{x_{j(1)}, ..., x_{j(k)}\}$ as a set containing the $k$ spatial closest points to $x_i$, $d_{sp}(i,j)$, the Euclidean distance spatially computed between adjacent pixels in
Chapter 5. Schroedinger Eigenmaps for Target Detection

\( N_k \), and \( \sigma_{sp} \), its corresponding spatial scaling factor. \( \gamma_{ij} \) and \( \eta_{it} \) are spectral scaling factors given in terms of the spectral angle (in radians) between the points \( \mathbf{x}_i \) and \( \mathbf{x}_j \) and between the target injected \( \mathbf{x}_t \) and its spectral neighbor \( \mathbf{x}_i \), respectively:

\[
\gamma_{ij} = e^{-\theta(\mathbf{x}_i, \mathbf{x}_j)} \quad \text{with} \quad \theta(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\mathbf{x}_i^T \mathbf{x}_j}{\|\mathbf{x}_i\|\|\mathbf{x}_j\|}\right),
\]

\( \eta_{it} = e^{-\omega(\mathbf{x}_t, \mathbf{x}_i)} \quad \text{with} \quad \omega(\mathbf{x}_t, \mathbf{x}_i) = \arccos\left(\frac{\mathbf{x}_t^T \mathbf{x}_i}{\|\mathbf{x}_t\|\|\mathbf{x}_i\|}\right), \]

Figure 5.5 shows a toy example showing how the magnitude of pixels in the Schroedinger embedding behaves as a function of the dimensions of the embedding, for the usual barrier potentials, i.e., the \( \mathbf{V} \) matrix, and for the barrier potential with the spatial-spectral knowledge propagation (SSKP), i.e., \( \mathbf{V}_{ss} \) matrix. The RGB rendering of the scene in Figure 5.5a shows two panels, one of blue color and the other one of red color, deployed on a gravel road that is surrounded by grass. In this example, the blue panel will be the material of interest, i.e., the target panel. Some interesting pixels are also highlighted with different markers: magenta dots represent the true target pixels, green circles are the spectral neighbors of the target signature previously injected to the data and the yellow squares represent their spatial neighbors, when a window of 4 pixels is considered. Since the detection rule for a test pixel is given as the inverse of its magnitude, as it will be shown later in this Chapter (see Section 5.3.2), the plots in the example show the magnitude for each one of the pixels in the considered scene as function of the number of Schroedinger Eigenmaps, i.e., the dimension in the lower dimensional space.

According to the theory behind the Schroedinger embedding, the labeled pixels in the barrier potentials and their neighbors will be pushed to the origin of the Schroedinger Eigenmaps. Then, looking at the magnitude, it is expected that the magenta lines, which are the magnitude for true target pixels, are flat lines at zero. Figures 5.5b-5.5g show the magnitude vs. Schroedinger dimension at different \( \hat{\alpha} \) values for normal barrier potential, in the first row, and for the spatial-spectral barrier potential (SSKP), in the second row; 20 dimensions in total are considered and only for distinction purposes, \( \hat{\beta} \) is used instead of \( \hat{\alpha} \) when SSKP is integrated to the barrier potential. Therefore, as for \( \hat{\alpha} \), \( \hat{\beta} = \hat{\beta} \frac{\text{Tr}(\mathbf{L})}{\text{Tr}(\mathbf{V}_{ss})} \).
Figure 5.5: Knowledge propagation example. (a) RGB rendering of a small scene with true target pixels, spectral and spatial neighbors highlighted. Magnitude as function of the dimensionality of the Schroedinger embedding at various $\hat{\alpha} / \hat{\beta}$ values. Black lines are the background, magenta lines are targets and green line is the injected target spectrum. (b) $\hat{\alpha} = 0.01$, (c) $\hat{\alpha} = 0.11$, (d) $\hat{\alpha} = 3.35$, (e) $\hat{\beta} = 0.01$, (f) $\hat{\beta} = 0.11$, and (g) $\hat{\beta} = 3.35$

Clearly, the plots in the first row show that even for the highest $\hat{\alpha}$ there are some magenta lines that start to increase in higher dimensions (around 18 SEs) and are mixed with the background lines in black color. This effect makes separating the targets from the background difficult. Contrary, in the second row, the plots show that the magenta lines are flatter at zero for higher $\hat{\beta}$ values. This kind of
behavior was the expected one according to the theory behind the Schroedinger embedding and this example shows how the addition of knowledge propagation to the usual barrier potential can help in the embedding and in turn, in the detection performance.

5.3 Methodology for Target Detection

Finally and combining the SE algorithm and the knowledge propagation scheme, an overall methodology for target detection could be proposed. The method is based on the SE algorithm described in Section 5.1 and incorporates the “knowledge propagation” scheme shown in Section 5.2. The overall detection algorithm includes two ways to reinforce the connectivity between the known target point and its neighbors in the potential. The first enforcing-connectivity approach is introduced in the creation of the spectral graph step, as it will be described below. The second technique takes advantage of the spectral-spatial knowledge propagation described in the previous section. In addition, as a way to model the variability of the target space, the method uses the measurements of target spectral signatures and to steer the Schroedinger embedding in desirable directions where the target-background separability is improved. It is assumed that different target signatures better encompass the spectral variability of the target space, therefore, the ideal scenario is to have multiple measurements of the target spectra. However, if only one target spectrum is available, the methodology can be adapted to use only that signature.

The input to the process is the hyperspectral image and the set of measurements of the target spectral signature. The image is re-arranged as a matrix of \(m \times n\), \(m\) pixels and \(n\) spectral bands; and because the target signatures are injected into the data matrix, the input will be a matrix of \([m + m_{ts} \times n]\), with \(m_{ts}\) as the number of measurements of the target signature added at the last positions of the input matrix. The algorithm’s steps are described as follow.

1. **Spectral Graph Creation.** A knn Graph on the whole input matrix is built using a global \(k\) (i.e., the same number \(k\) is used in the searching for the nearest neighbors of each pixel), and Equation (3.6) is used for computing the weight matrix, \(W\) but with the “Spectral Angle distance” (SAD) as spectral distance. The use of spectral angle is based on the fact that its response, unlike the Euclidean distance, highlights the difference in shape between two vectors over the difference in magnitude. The spectral angle is given by

\[
d_{SA}(x_i, x_j) = \arccos \left( \frac{x_i^T x_j}{\|x_i\| \|x_j\|} \right)
\]

(5.10)
In addition, the spectral connectivity for the injected target signature is redefined in the sense of having in its neighborhood, pixels with edge lengths ≤ 2. This redefinition of the target signature connectivity is made with the idea to enforce the target connectivity with pixels that share some similarity, but not enough to be directly connected.

2. **Spatial Graph Creation.** A knn Graph with non-zero entries at the spatial neighbors of the target’s spectral neighbors (sNN) is created. Spatial windows of 4 or 8 pixels around each spectral neighbor (considering edge lengths ≤ 2) of the target signature are considered. The spatial weight matrix, $W_{sp}$, is computed using Equation (5.7) in Section 5.2.

3. **Combining spatial-spectral Graphs.** The spectral and spatial knn graphs are combined as $W_{ss}$, by adding the spatial and spectral weighted matrices.

4. **Laplace Operator.** The weighted degree matrix for the spectral knn graph is computed as an intermediate step for computing the Laplacian matrix, $L$ (where the Laplacian only considers the spectral information).

5. **Barrier Potentials.** The diagonal matrix $V$, corresponding to the barrier potential, is defined as having on the main diagonal non-zero entries, i.e., ones, at the location of the spectral neighbors of the target signature, and at the position of the target signature itself.

6. **Knowledge propagation matrix.** The modified-barrier-potential matrix $V_{ss}$, that includes the spectral and spatial connectivity of the barrier potentials is computed using Equation (5.5) in Section 5.2.

7. **Schroedinger Operator.** The Schroedinger matrix, $S$, is computed by adding the modified-barrier potential matrix to the Laplacian

$$S = L + \beta V_{ss}. \quad (5.11)$$

The modified-barrier potential is weighted by the scalar factor $\beta$ that similar to $\alpha$ previously, is modified as $\beta = \hat{\beta} \times \frac{\text{Tr}(L)}{\text{Tr}(V_{ss})}$ with $\hat{\beta}$ varying in a [0, 100] range of values. $\hat{\beta}$ is introduced as a scaling factor that is independent of the data that is processed.

8. **Eigendecomposition.** The eigenvectors, $\Psi$, and eigenvalues, $E$, of the $S$ matrix are found and arranged in ascending order, i.e., smallest eigenvalues
are considered first. Then, the set of the $\ell$ first eigenvectors are kept as the lower dimensional representation of the data. The number $\ell$ of eigenvectors that normally are called Eigenmaps, is estimated by using the Gram matrix method [66] applied to the transformed data.

9. Detection Operator. The detection is performed by taking advantage of the fact that in the Schroedinger embedding, the barrier potential constraints and the nearby points are pulled together near the origin as a results of the transformation. In this way, the detection operator for each point $i$ is given by

$$T_{SEi} = \frac{1}{\|\Psi_{SEi}\|}. \quad (5.12)$$

Figure 5.6 shows the flowchart explaining graphically the overall methodology and the links between the different steps. Three steps in the overall algorithm play a key role in the task of improving the distinction and separability of the target class against the background. These are: the creation of the spectral graph, the spatial-spectral combination by using a Knowledge Propagation scheme and the definition of the target detector. The use of the knowledge propagation for combining spatial-spectral information was previously explained in Section 5.2. The other two steps are explained in more detail below, with the idea of having a complete picture of how the methodology works.

Figure 5.6: Flowchart of the overall target detection methodology based on SE and Knowledge Propagation.
5.3.1 Spectral Graph Creation

Throughout this research has been noted that the overall performance of the SE-based methodology for target detection, depends to some extent on how well the local structure of the data is represented. In our methodology, the creation of the spectral graph is largely related to the structure representation, but also the construction of the spatial graph (despite only for some points the spatial connections are considered) plays an important role in this matter. A detailed explanation about the approaches proposed for building the spectral and spatial graphs will be presented below in the next paragraphs.

For the spectral graph, the typical creation based on the $k$ searching technique is modified in two ways:

- Two approaches for estimating the $k$ parameter are considered: **Global knn Graph** and **Adaptive knn Graph**.

- The connectivity of the target space is defined as keeping pixels with $\leq 2$ edges as neighbors of the target spectral signatures injected into the data.

**Estimation of $k$ for knn Graph creation.** Once the target spectral signature(s) is(are) injected into the image, the knn Graph is created by following two approaches. Both approaches use the commonly known knn searching technique for finding the nearest neighbors for each pixel. The difference between the approaches is the estimation of $k$. The first approach uses a global $k$ number such that the same $k$ nearest neighbors are searched for each pixel in the data set. The use of a single $k$ introduces the issue of estimating an appropriate $k$ value that reconstructs as best as possible the local structure of the data. Since there is no rule that indicates an appropriate $k$ value, the choosing of $k$ is generally done by a trial and error approach. It seems suitable to use an alternative way for estimating different $k$ values for different pixels depending on the characteristics of the local distribution or spread of the data pixels.

In this way, the second approach that is considered in this research, is the estimation of an adaptive $k$, one that uses a different $k$ value depending on the local regions of the image. The estimation of $k$ is performed in an adaptive fashion such 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, i.e., those points have few connections. Therefore, few neighbors are assigned to pixels in spectral regions with the lowest density, and a maximum $k$ neighbors to those with the highest density. The adaptive estimation of $k$ is introduced as a way to minimize
Chapter 5. Schroedinger Eigenmaps for Target Detection

the impact of pixels outside clusters of the data set.

This technique was initially proposed by Mercovich, et al. (2011) [59] as method to represent spectral imagery using graphs. The process includes the computation of a co-density score 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 $k_{\text{max}}$ neighbors divided by $k_{\text{max}}$, where $k_{\text{max}}$ is set by the user and usually is a value in a range of [60, 80]. Then, the distribution of co-density measurements over the data sets is divided into segments and the pixels that fall into the same segment will have the same $k$ value, which will be different from the $k$ values for other segments of the distribution. According to Mercovich, et al. and from experimental results, the co-density distribution is generally well represented by a normal distribution and the number of bins that gives good results is empirically set to 6. The bins will be defined by segmenting the distribution based on the standard z-scores, where the lowest threshold is set to the z-score of $k-2$ and the highest to z-score $k+2$, and the values for $k$ are defined experimentally. In our case, the $k$ values are defined in terms of the $k_{\text{max}}$ times the minimum value of the co-density score over the maximum co-density value in each bin. This computation was chosen just considering a constant variation for $k$ through the segments. Figure 5.7 is a graphic description of this technique.

Figure 5.7: Flowchart of the adaptive $k$ estimation technique for the graph creation.

Figure 5.8 shows the same example described in Section 3.2.1 but in this case the use of the adaptive $k$ estimation is added. It can be observed in Figure 5.8d 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, i.e., those points have few connections. Although the two classes, the black and the pink, are connected in the three cases considered in this example, the adaptive estimation of
Chapter 5. Schroedinger Eigenmaps for Target Detection

$k$ produces only one edge connecting the 2 clusters, which probably can be managed better later in the next steps of a classification or target detection approach.

![Figure 5.8: 2-D Synthetic data set.](image)

(a) Original data set is a sum of Gaussians with different means and equal covariances. (b) knn Graph with $k = 2$. (c) knn Graph with $k = 8$. (d) knn Graph with Adaptive $k$

The second part in the creation of the spectral graph is related to the connectivity of the target space. In general, the definition of neighborhood in the graph-related terms is conformed by the pixels that are connected to a specific pixel or point of interest. In this case, the neighborhood of the target signature(s) is re-defined as considering not only the pixels directly connected to the target signatures(s), but also the neighbors of the neighbors, i.e., pixels of connection length $\leq 2$. These new connections for the target are introduced with the idea of improving the connectivity of the target signature(s) with pixels which share some similarity,
but not enough to be directly connected. Figure 5.9 shows step by step the process that is followed to build the spectral graph based on the new definition of the target neighborhood, and also how the spatial graph is defined.

Figure 5.9: Toy example for explaining the spectral and spatial Graphs creation. Target signatures are shown in gray color, actual target points in the data set are in magenta color, and the background points in blue color. (a) Data in spectral domain. (b) The spectral kNN Graph built on the data. (c) Identification of target signature neighborhood, sNN, considering edge lengths ≤ 2. (d) Over-connection of the target cluster – target signatures and their sNN points. (e) Spatial Graph for sNN points.

As an important part for creating a weighted graph is the weighting function. Equation (3.6) is widely used for this purpose. However the equation has a σ parameter that generally is set to one single empirical value. The issue with this method to define σ is that one single value for σ may not work well for all pixels in an image, especially in cases where the data set has multiple scales, and the local statistics
generate different local data distributions. Therefore, an adaptive local sigma concept is introduced in this research, with the idea of estimating different σ values along the image depending on the local distribution of the data set. It was initially proposed for clustering by Zelnik and Perona (2004) [67], but it is directly applied to our problem here since its aim is to better capture the local structure of the data set. Figure 5.10 attempts to show the difference between use a single value of σ and use several values depending on the data local structure.

Figure 5.10: The impact of using a single σ or local σ values. (a) Toy Data Set of two clusters with different local distribution. (b) Affinity between data points using a single σ value. (c) Affinity between data points using local σ values. The thickness of the edges represent the affinities between each point and its neighbors.

The example in Figure 5.10 is a toy data set consisting of two clusters with different local distributions and the affinities between each point and its neighbors are indicated by the thickness of the edges. Figure 5.10b shows that the affinities across the two clusters are larger than some affinities within the clusters. Meanwhile, in Figure 5.10c the affinities across clusters are much lower than the affinities within any of the two clusters. Therefore, a local scaling method results in high affinities within cluster and low ones across them. In addition this kind of approach could automatically find the appropriate scales of the data set and consequently improve its representation. The method calculates a local σ for each data pixel \( x_i \), so that the distance from \( x_i \) to \( x_j \) is given by \( \frac{d(x_i, x_j)}{\sigma_i} \) and the converse, i.e., the distance from \( x_j \) to \( x_i \), is \( \frac{d(x_j, x_i)}{\sigma_j} \). Thus, the square distance \( \frac{d^2_{ij}}{\sigma^2} \) in Equation (3.6) could be written as

\[
\frac{d^2_{ij}}{\sigma^2} = \frac{d_{ij}d_{ji}}{\sigma_i\sigma_j} = \frac{d^2_{ij}}{\sigma_i\sigma_j}.
\]
Therefore the weight matrix that is going to be used in this methodology is rewritten as

\[
W_{ij} = \begin{cases} 
  e^{-\frac{d_{ij}^2}{\sigma_i \sigma_j}} & \text{if } G_{ij} = 1, \\
  0 & \text{otherwise.}
\end{cases} \tag{5.14}
\]

The estimation of the local \(\sigma_i\) and \(\sigma_j\), in turn is based on the local statistics of the neighborhood of each pixel \(x_i/x_j\). In this case, the distance to the k nearest neighbor is considered such that \(\sigma_i = d_{i,k}\), where \(k\) is the number of nearest neighbors of \(x_i\), and it is manually set for the user or it could be estimated by the weighted density knn method.

### 5.3.2 SE-based Detection Definition

As it was previously stated, the Schroedinger transformation has the property of pulling barrier potentials and their nearby pixels toward the origin of the Schroedinger embedding. This fact is taken into consideration for defining a detection rule where those barrier potentials and the nearby pixels have higher values than those pixels distant from the origin. Therefore, the detector is given in terms of the inverse of the Euclidean distance of each pixel (to zero) in the Schroedinger embedding:

\[
T_{SE_i} = \frac{1}{\|\Psi_{SE_i}\|}, \tag{5.15}
\]

where \(\Psi_{SE_i}\) is the vector of values in the \(\ell\) dimensional space for the pixel \(i\), and \(\|\cdot\|\) is the Euclidean norm. In other words, \(\|\Psi_{SE_i}\|\) is the magnitude of pixel \(i\) in the Schroedinger embedding. This detection rule implies a circular decision boundary as it is shown in Figure 5.11. The example shows a toy data set comprised by two normal distributed classes shown in magenta and black colors. The figure also shows the Schroedinger embedding where the magenta points are the class of interest, and the decision boundary derived from Equation (5.15). The magenta points within the circle and the blue points are out of the circle, i.e., points within the circle are categorized as the target class and the points out of the circle corresponds to the background.
Figure 5.11: Decision boundary of the detector. (a) Toy Data Set of two normal distributed classes drawn in magenta and blue colors. (b) The Schroedinger embedding in a 2D space with the circle drawing the decision boundary.

5.4 Summary

A SE-based methodology for spectral target detection has been presented in this Chapter, including the description of the SE algorithm and the knowledge propagation scheme. The Chapter is divided into three sections. The first section is related to the SE algorithm, where definitions and mathematical variables involved in the Schroedinger operator are presented. The explanation and supportive concepts about the optimal Schroedinger embedding are described as well as the difference with the Laplacian embedding. The second section introduces the knowledge propagation scheme, initially proposed for clustering applications, with modifications for combining spatial and spectral constraints in order to improve the identification of the target class. In the last section, the overall SE-based methodology for target detection is described, including details of its implementation, particularly for the main steps of the algorithm that largely impacts the embedding. The next chapter will present the results after applying this SE-based target detection methodology to different hyperspectral data sets.
Chapter 6

Graph and SE-based Target Detection

The target detection methodology based on Schroedinger Eigenmaps (SE) with the adaptation of a knowledge propagation scheme, which was described in Chapter 5 is applied to different hyperspectral (HSI) data sets. The performance of the overall methodology is assessed from a qualitative point of view by visually analyzing the detection images. Additionally, a quantitative analysis is carried out by using ROC curves and the rates of detection and false alarm extended to the object detection, described in Chapter 2. In addition, a preliminary experiment testing the applicability of graph-based methodologies in target detection is shown at the beginning of this Chapter. This experiment was performed as an initial exploration of the utility of graph-based methodologies in target detection, given most of these types of methods have been widely applied to classification related problems, but there still is a lack of testing of how these methods work in target detection.

6.1 Graph-based Target Detection

As an initial step in the study of the target detection problem in HSI using non-linear manifold learning methods, the use of graph-based algorithms was explored. In this way, the Topological Anomaly Detection (TAD) algorithm was modified and used for target detection. This algorithm (mentioned in Chapter 3) is a graph-based method developed for anomaly detection. It has as a primary tool the representation of connectivity between data points by using an adjacency graph. The overall methodology includes the creation of a knn graph made up of multi-
ple connected components that mainly form the background, and the calculation of anomalousness scores for each pixel. The result is a map of TAD scores, where pixels with higher scores have an increased anomalousness [45]. The idea to carry out this study was to explore the use of TAD as a means of, and alternate way to, characterize the image background components and use that model as background basis in “traditional” detectors. The TAD-based methodology for target detection includes a background characterization by following different approaches, depending on the type of detector that is used. Two detectors, one based on a statistical model and the other in geometric model, are used: Adaptive Coherence Estimator (ACE) and Orthogonal Subspace Projection (OSP). The steps are described as follows:

1. **TAD background model.** This step is completely based on the TAD algorithm. Initially, the entire hyperspectral image is normalized such that the data is scaled to lie between two nested hyperspheres with radii of $r = 1$ and $r = 2$ respectively. Then a random subsample of pixels is chosen with the purpose of modeling the background of the scene by using that random sample. The sample size is typically between 500 to 10000 pixels, and it is assumed that all major background components are represented in the sampling. After this, a knn graph is built by using the knn searching technique, where a global $k$ number is empirically set. The spectral Euclidean distance is also used to find the $k$ nearest neighbors and to weight the edges. However, only the edges between the 10% closest pairs of pixels are preserved. Once the adjacency graph is built, some clusters of connected pixels can be noted. The largest components of connected points containing greater than 2% of the sample size are designated as background.

2. **Background characterization.** Once the background connected components are estimated, different approaches based on the two perspectives, geometric or statistical, are proposed to characterize them. Five different approaches that depend on the type of detector are considered: three approaches for geometric detectors and two approaches for the statistical ones. **Geometric approaches** use the mean of each component as the background basis in the detectors, the endmembers of components as background basis for each background component, such that there are as many detection maps as number of connected components, and all endmembers estimated for all components are used together as background basis vectors in the detectors. The number of endmembers and the endmembers were estimated by using the Gram Matrix approach [66], and the MaxD algorithm, respectively. For the **Statistical approaches**, there are only two ways to characterize the TAD background
components. These approaches include the estimation of the mean-vector and covariance-matrix for each component, and for all of pixels, where all of the pixels designated by TAD as background were used to estimate the mean-vector and covariance-matrix.

3. Detection. After the background model and characterization, the detection is carried out using two detectors: OSP and ACE. OSP is used in those cases where the characterization of the background was based on the geometric perspective, whereas ACE is used in the approaches based on the statistical perspective.

The assessment of this TAD-based methodology applied to target detection was carried out by using the self test data that is part of the RIT Target Detection Blind Test project [68] and it is available online [69]. The self test data includes complete ground truth information, target spectral signatures from laboratory and field measurements, and their locations. Figure 6.1 shows the RGB rendering indicating the location of the target panel of interest, and its spectral signature measured in the field. The fabric panel target is labeled as F1 in the project and it has 1 fully resolved pixel, 8 sub-pixels and 16 guard pixels.

Figure 6.1: Self test data set. (a) HyMap Image including target F1. (b) Field measurement of F1 spectral signature.
Chapter 6. Graph and SE-based Target Detection

For the TAD background model step, 2500 pixels as subsample and 4 nearest neighbors were used in the graph creation. The TAD outcome was three background components with 54, 1319 and 433 pixels. Figures 6.2 and 6.3 show the detection maps and their histograms for the OSP detector and for the different approaches. In each histogram, the fully resolved target and the sub-pixel targets are marked in green and red lines respectively indicating how far the target pixels are from the background. An additional experiment was performed in order to compare its results with the TAD-based results. This experiment included the estimation of endmembers for all of the pixels in the whole image. Their results are noted as Global Approach.

Figure 6.2: OSP Maps and histograms. (a) Approach using means. (b) Approach using EMs for 1st component. (c) Approach using EMs for 2nd component.
Figure 6.3: OSP Maps and histograms. (a) Approach using EMs for 3\textsuperscript{rd} component. (b) Approach using all EMs together. (c) Global Approach.

These OSP results show that the better results are obtained for the 3\textsuperscript{rd} component, the fully resolved target pixel is well separated from the background (see Figure 6.3b), and even some sub pixel targets are near the fully resolved target. For the global approach (see Figure 6.3c), the histogram is wider than the other five cases and the target pixels are close to the background. Figure 6.4 shows the results for the ACE detector. In this case, only the 2\textsuperscript{nd} and 3\textsuperscript{rd} components are considered, since for the 1\textsuperscript{st} component the number of pixels is less than the number of spectral bands. In the last case, the covariance estimation is unstable, and the covariance matrix becomes singular, meaning that it can not be inverted (see definition of statistical detectors in Chapter 2, the inverse of covariance matrix is required). Also the Global approach is used it as a means to compare TAD-based results.
Figure 6.4: ACE Maps and histograms. (a) Approach/all components together. (b)-(c) Approach/each component 2\textsuperscript{nd} and 3\textsuperscript{rd} components respectively. (d) Global Approach.

The histograms use the same code of color for differentiate the full-pixel target, subpixel target and the background pixels. The histograms for the ACE detection cases show that the fully resolved target pixel is far from most of the scene’s pixels,
which will facilitate setting a detection threshold. For the global approach, there is not much difference between its results and the results using TAD background components. These results show that ACE achieved a better performance than OSP, imply statistical models have better performance when using the TAD-derived background components. Based on ACE results, it is possible to say that graph-based methods, specifically TAD, have a great potential in target detection problems. A more wide study using other two detectors, Constrained Energy Minimization (CEM) and Adaptive Subspace Detector (ASD) is presented in [49].

6.2 SE-based Target Detection Applied to HSI

The SE-based target detection method described in Chapter 5 is applied to three HSI data sets that are described below. Three different materials are considered as target material and for some of them there are several distinct target spectra available. Each image and material are analyzed by using three methodologies: SE-based method with spatial-spectral knowledge propagation using a global $k = 20$ (SE-SSKP; $k = 20$), SE-base method with spatial-spectral knowledge propagation using adaptive $k$ (SE-SSKP; Adpk), LE Algorithm and ACE detector (LE; ACE). In addition, the methods are compared against the ACE detector in the uncorrelated spectral domain (PCA; ACE). In addition, an experiment to estimate an appropriate $\hat{\beta}$ parameter is presented here.

6.2.1 HSI Data sets

1. SHARE 2010: This data set was collected as part of a data campaign hosted by Rochester Institute of Technology (RIT) over the RIT campus in 2010 [70]. The data set is in reflectance and include hyperspectral images and target spectral signatures measured in the lab and in the field. The hyperspectral pushbroom spectrometer ProSpecTIR VS was used in this campaign to capture the hyperspectral images. The sensor covers the $0.4 - 2.4 \mu m$ range with 360 bands and has a spatial resolution of approximately 1m. The image considered in this research has 86 x 146 pixels and 328 bands after absorption bands are removed. The image is an urban scene containing several materials or objects such as asphalt, brick, vegetation, concrete, roof, buildings, cars, parking lots, etc. Also, blue and red cotton felt panels in 2m x 2m and 3m x 3m sizes are included in the scene at known locations. The panels are located at areas with different illumination and occlusion conditions. Figure 6.5 shows two RGB renderings with different contrast enhancement showing the
locations of the target panels framed in magenta squares. The RGB rendering with
the 2% linear contrast enhancement in Figure 6.5 allows one to see most of the
target panels, however, there are some blue and red panels located in shadows that
are visually missed. The RGB rendering with the 255 linear contrast enhancement
in Figure 6.5b shows those panels in shadows although most of the values are over
saturated.

Figure 6.5: SHARE 2010 showing blue and red panels locations. (a) RGB rendering
with a 2% linear enhancement. (b) RGB rendering with a 255 linear enhancement.

2. SHARE 2012: This data set is also part of a data campaign hosted by RIT in
September of 2012 and conducted in the AVON, NY area. Multiple sensors were used
and different experiments were conducted for specific purposes [71]. The collected
data as well as their ground truth are available online [72]. One of the experiments
was related to the acquisition of hyperspectral ground-truthed data set for target
detection. The sensor is again in this case, the same ProSpecTIR VS spectrometer
used in SHARE 2010. The experiment included the deployment of panels with
various sizes in areas with various levels of illumination and occlusion, so as to have
targets with different complexity of detection. The same red and blue felt panels
used in SHARE 2010 were used in this experiment. The image in this case has a size
of 270 x 180 pixels and 292 spectral bands after removing the absorption bands. This
scene in Figure 6.6 includes only some of the total deployed target panels. Figure
6.6a shows an enhanced RGB rendering that includes the target areas highlighted
by magenta squares. Figures 6.6b and 6.6c show some of the available target spectra
for the blue and red panels. The data set includes multiple field measurements of the blue and red spectra; there are 8 and 9 spectra for the red and blue felt panels. Figures 6.6b and 6.6c show four blue and red spectra, respectively. Two of them (in each figure) correspond to field measurements taken from target panels located at open field and in shadows, and the other two are in-scene spectra that correspond to target pixels in the open field and in shadows. These blue and red spectra are displayed as an attempt to show the target spectral variability as measured and the actual spectral variability of the target materials in the image.

Figure 6.6: SHARE 2012 showing blue and red panels locations and target material spectra. (a) RGB rendering with a 2% linear enhancement. (b) Blue target spectra. (c) Red target spectra.

3. MUUFL campus: This scene is part of a data collection conducted in Novem-
ber 2010 covering the campus of the University of Southern Mississippi - Gulfport, located in Long Beach, Mississippi. Hyperspectral and LIDAR data are included in the collection and are available online [73]. The hyperspectral sensor was the CASI-1500 that has 72 bands covering the visible and part of near infrared (0.375 – 1.050µm), and a spatial resolution of approximately 1m. The target detection experiment includes 60 cotton fabric panels of three different sizes: 0.5m x 0.5m, 1m x 1m and 3m x 3m, so there are 20 panels of each size and of those 20 panels, four different color fabrics were used. Therefore, each color cover 5 panels of the same size, making 15 panels for each color fabric. In this case one specific fabric color is considered as target material, this is the “Pea green” target panel.

Figure 6.7: MUUFL campus showing Pea green targets locations and target material spectra. (a) RGB rendering with a 2% linear enhancement showing in magenta the sub-pixel targets and in yellow the full pixel targets. (b) Pea green target spectral signature measured in the field and an in-scene pea green target spectrum.

The locations of the pea green panels are highlighted by magenta squares in Figure 6.13 the field pea green spectrum and the corresponding in-scene pixel spectrum are shown in Figure 6.7b. Given the sizes of the panels, only the 5 pea green fabric panels of 3m x 3m guarantee at least 2 or 3 pure pixels but given their locations in the scene (see yellow squares in Figure 6.13), most of them are occluded by tree coverage which imposes greater difficulty in their detection. The other two sizes that are smaller than the spatial resolution of the sensor will establish a sub-pixel
6.2.2 Finding an appropriate $\hat{\beta}$

According to the discussion of SE algorithm in Chapter 5, the $\alpha$ parameter, or the $\beta$ parameter after the inclusion of “knowledge propagation”, plays an important role in the Schrödinger embedding. The experiment attempts to address the issue of how to choose an appropriate value for the $\beta$ parameter (see Eq. 5.11). Figure 6.8 shows the flowchart of the algorithm followed to test the impact of $\hat{\beta}$, and based on the results to set an appropriate $\hat{\beta}$ value. The idea is to vary the value of the parameter $\hat{\beta}$ in Eq. 5.11 in a range from 0 to 100, and to compute the detection map as a function of the number of eigenvectors or Schrödinger Eigenmaps (SEs) as they are commonly known (see Eq. 5.15). The experiment is designed such that the maximum number of SEs that allows the target detection with a minimum FA rate (FAR) is computed for each $\hat{\beta}$ value. FAR is based on the calculation of the FA score (FAs), which is the metric for determining the number of false alarms. As it was explained in Chapter 2, the FAs is useful in cases where the total number of target pixels is much smaller than the total number of pixels in the scene, and gives information about how many false alarms are obtained when a given target pixel is detected for a determined detection map. Different target pixels could be chosen as the target value to be detected, for example, the target pixel with the largest value in the detection map (i.e., the easiest target pixel). Here, the maximum value of all the target pixels in a given detection map is considered. Therefore, the appropriate value of $\hat{\beta}$ for each considered data set will be the value that gives a maximum number of SEs for being used in the detection with a minimum of FAR.

The experiment is carried out for the three data sets, SHARE 2010, SHARE 2012, and MUUFL shown in Figures 6.5-6.7, the number of SEs used in the detection with a minimum of FAR = 0.001 are shown in Table 6.1. According to the table, there are variations on the number of SEs that can be used in the detection when the $\hat{\beta}$ is varied. There is not a clear tendency in the values for the number of SEs, however, it is identifiable a $\hat{\beta}$ value in which the number of SEs is maximum. These are the $\hat{\beta}$ values used in the detection experiments present below. These values are highlighted in bold in the Table 6.1 and they are 0.11 for SHARE 2010, 0.01 for SHARE 2012, and 0.7 for MUUFL.
Figure 6.8: Flowchart for the experiment testing $\hat{\beta}$ value vs. the number of SE that can be used in the detection with a $\text{FAR} \leq 0.001$.

Table 6.1: Number of Schroedinger Eigenmaps at different $\hat{\beta}$ values

<table>
<thead>
<tr>
<th>Data Set</th>
<th>$\hat{\beta}$ values</th>
<th>0.001</th>
<th>0.01</th>
<th>0.11</th>
<th>0.7</th>
<th>1</th>
<th>5</th>
<th>14</th>
<th>37</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. SHARE 2010</td>
<td></td>
<td>66</td>
<td>66</td>
<td>73</td>
<td>59</td>
<td>57</td>
<td>40</td>
<td>33</td>
<td>61</td>
<td>67</td>
</tr>
<tr>
<td>2. SHARE 2012</td>
<td></td>
<td>24</td>
<td>32</td>
<td>23</td>
<td>21</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>11</td>
<td>20</td>
</tr>
<tr>
<td>3. MUUFL campus</td>
<td></td>
<td>9</td>
<td>24</td>
<td>24</td>
<td>18</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>11</td>
<td>11</td>
</tr>
</tbody>
</table>

Note. - The number of SEs summarized here are the dimensions in the Schroedinger domain that are considered in the detection and that gives a minimum FAr ($\text{FAR} = 0.01$). The SEs are computed for different $\hat{\beta}$ values and for the three hyperspectral data sets. The results for SHARE 2010 and 2012 corresponds only for blue material.

6.2.3 Target detection results for SHARE 2010

The experiments for SHARE 2010 include the detection of blue and red materials, red cotton felt panels and blue cotton felt panels. As it was established at the beginning of this Chapter, four approaches for detecting the targets are consid-
Chapter 6. Graph and SE-based Target Detection

Considered: the two based on the Schroedinger Eigenmaps algorithm, SE-SSKP, $k = 20$ and SE-SSKP, Adp $k$, one using Laplacian Eigenmaps algorithm and ACE detector, LE-ACE, and the other using the linear transformation method PCA and ACE detector, PCA-ACE. All target spectral signatures measured in the field and that are available, are used as a way to encompass a wide spectral variability of the target space. The detection results are analyzed from a visual and quantitative point of view. Detection maps, which are the images generated by the detectors and whose values represent how probable it is that each pixel belongs to the target class or background class, are displayed in a grayscale colormap where whiter pixels have a greater probability to be target. Their binary versions are generated after applying a threshold that is set to the top 0.8\% of the total pixels. This is, the first 98 pixels (total pixels= 12264) with higher values in the detection map will have value 1 in the binary map. From the quantitative point of view, ROC curves are calculated for each detection approach.

For the blue material, the pair of images, detection maps and binary maps, corresponding to each detection technique are shown in Figure 6.9, as well as, the ROC curves in Figure 6.9b. The estimated dimensionality for each domain was determined to be 3 for the Schroedinger Eigenmaps using $k = 20$, 4 for Schroedinger Eigenmaps using Adaptive $k$ ($k_{max} = 60$), 4 for Laplacian Eigenmaps and 4 for the PC embedding. The detection and binary maps show that almost all the target panels are identified even one of those in the shadows, representing a target with a difficulty of detection. Looking at the ROC curves, both SE-based techniques have the highest detection rates at lowest false alarm rates, i.e., these two methods outperform the ACE detector in the Laplacian and in the PC domains. Regarding LE-ACE and PCA-ACE approaches, pixels in buildings’ roof and shadows are the brightest ones, and the target panel areas exhibit darker tones, from which it is deducible that the detection of the blue panels involves a high number of false alarms. This behavior is reflected in the ROC curves, where high rates of detection implies also high rates of false alarms. LE-ACE technique outperforms PCA-ACE approach at least at the lowest false alarm rates. Nevertheless, PCA-ACE outperforms LE-ACE at moderate and highest false alarm rates.

For the red material, detection maps, binary maps and ROC curves are shown in Figure 6.10. The estimated dimensionality in each domain are: 4 for the Schroedinger embedding when $k = 20$ is used, 4 for the Schroedinger embedding using Adaptive $k$ ($k_{max} = 60$), 4 for the Laplacian embedding and 4 for the PC embedding. Comparing these results against the results for the blue material, there is a consistency
Figure 6.9: SHARE 2010 detection results for blue material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and \( k = 20 \), (d) SE-SSKP and Adaptive \( k \), (e) LE-ACE, (f) PCA-ACE.
in the results. The trend where SE-based methods outperform the other two approaches continues. However, the rates of detection for SE-based methods at lowest false alarm rates are lower, which means there are more false alarms when the rates of detection are highest. As to LE-ACE and PCA-ACE techniques, they have almost the same performance at lowest rates of false alarms, but at high rates of false alarms, LE-ACE technique outperforms PCA-ACE. In addition, the rate of object detection, $r_{OD}$ described in Chapter 2 is used as an alternative way to evaluate the techniques if the detection is seen as a problem where the most important is the identification of the target area and not to detect each of the pixels that conform the target.

Table 6.2 shows the detection rates but also a modified false alarm rate, $r_{FOA}$ whose calculation is achieved after removing all pixels that are part of the whole target-objects detected and reported in the $r_{OD}$. The rates are computed at the same threshold used for the visual results in Figures 6.9 and 6.10. For both materials, SE-SSKP with $k = 20$ has the best detection performance with the lowest false alarm rate possible, even for blue material, the object detection rate is perfect with a very small $r_{FOA}$. Although for red material there is no perfect detection, the two SE-based techniques have the highest rate of detection with the lowest rates of false alarm.

Table 6.2: SHARE 2010 results for blue and red material

<table>
<thead>
<tr>
<th>Detection Technique</th>
<th>Blue material</th>
<th>Red material</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$r_{OD}$</td>
<td>$r_{FOA}$</td>
</tr>
<tr>
<td>SE-SSKP; $k = 20$</td>
<td>1</td>
<td>$3.36 \times 10^{-3}$</td>
</tr>
<tr>
<td>SE-SSKP; Adpk</td>
<td>0.86</td>
<td>$4.26 \times 10^{-3}$</td>
</tr>
<tr>
<td>LE-ACE</td>
<td>0.29</td>
<td>$7.95 \times 10^{-3}$</td>
</tr>
<tr>
<td>PCA-ACE</td>
<td>0.14</td>
<td>$8.03 \times 10^{-3}$</td>
</tr>
</tbody>
</table>

Note. - Object-detection rates and False Object rates for blue and red material at 0.8% of total number of pixels (np) as thresholds.
Figure 6.10: SHARE 2010 detection results for red material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) $\text{SE-SSKP and } k = 20$, (d) $\text{SE-SSKP and Adaptive } k$, (e) LE-ACE, (f) PCA-ACE.
It has been noticed that SE-SSKP with \( k = 20 \) approach outperforms the SE-SSKP with adaptive \( k \) approach. However, if another values for the parameter \( k \) are considered such as \( k = 4 \), the global \( k \) approach does not outperform the adaptive \( k \) as it was shown in [73]. Therefore, the performance of the SE-SSKP approach with a global \( k \) varies depending on the value for \( k \). For example, if two points are far away from each other, the global \( k \) technique with a small \( k \) will probably decide they are not connected, i.e., the connecting edge is 0; instead, if \( k \) is large, the approach will decide the two points are connected by an edge with a low weight. In this sense, it is much better that the two points are connected by a weak edge than they do not have an edge at all since the two points probably are not equally far away [75]. This balance in the choosing of \( k \) could be problematic if there is no idea of what value for \( k \) is small or large, in those cases the adaptive \( k \) approach could be helpful.

### 6.2.4 Target detection results for SHARE 2012

These experiments also include the detection of blue and red felt panels and the analysis of the four detection techniques, SE-SSKP with \( k = 20 \) and SE-SSKP with Adaptive \( k \), LE-ACE, and PCA-ACE. Detection maps are again displayed in a grayscale colormap and for the binary versions, the threshold is set to the top 0.8% of the total pixels (\( np = 47600 \)).

**Blue material** results are shown in Figure [6.11]. The estimated dimensionality in each domain is: 4 for Schroedinger Eigenmaps using \( k = 20 \), 4 for Schroedinger Eigenmaps using Adaptive \( k \) (\( k_{\text{max}} = 60 \)), 2 for Laplacian Eigenmaps and 6 for the PC embedding. The detection and binary maps show that almost all the target panels are identified even one of those in the shadows, a particularly difficult target to detect. Looking at the ROC curves, both SE-based techniques have the highest detection rates at lowest false alarm rates, i.e., these two methods outperform the ACE detector in the Laplacian and in the PC domains. Regarding LE-ACE approach, pixels in shadows are the brightest ones, while for PCA-ACE approach, pixels in the forest area have high values (they are the brightest pixels); besides, target panels in the open grass field exhibit darker tones. From these reasons it is deducible that the detection of the blue panels involves a high number of false alarms. This behavior is reflected in the ROC curves, where high rates of detection implies also high rates of false alarms, and LE-ACE technique outperforms PCA-ACE approach.
Figure 6.11: SHARE 2012 detection results for blue material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE.
Figure 6.12: SHARE 2012 detection results for red material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE.
For the red material, detection maps, binary maps and ROC curves are shown in Figure 6.12. The estimated dimensionality in each domain are: 4 for both Schroedinger Eigenmaps with $k = 20$ and Adaptive $k$ $(k_{max} = 60)$, 3 for Laplacian Eigenmaps and 6 for the PC embedding. Comparing these results against the results for blue material, there is a consistency in the results, the trend where SE-based methods outperform the other two approaches continues. However, the rates of detection for SE-based methods at lowest false alarm rates are lower, which means there are more false alarms when the rates of detection are highest. As to LE-ACE and PCA-ACE techniques, they have almost the same performance at lowest rates of false alarms, but at certain range of false alarm rates, LE-ACE technique outperforms PCA-ACE and in another range, the contrary happens. The rate of object detection, $r_{OD}$, and the rate of false object-alarms, $r_{FOA}$, at four different thresholds are shown in Table 6.3.

<table>
<thead>
<tr>
<th>Detection Technique</th>
<th>Blue material</th>
<th>Red material</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$r_{OD}$</td>
<td>$r_{OD}$</td>
</tr>
<tr>
<td>SE-SSKP; $k = 20$</td>
<td>1</td>
<td>7.06 $\times 10^{-3}$</td>
</tr>
<tr>
<td>SE-SSKP; Adpk</td>
<td>1</td>
<td>7.04 $\times 10^{-3}$</td>
</tr>
<tr>
<td>LE-ACE</td>
<td>0.14</td>
<td>8.01 $\times 10^{-3}$</td>
</tr>
<tr>
<td>PCA-ACE</td>
<td>0.14</td>
<td>8.03 $\times 10^{-3}$</td>
</tr>
</tbody>
</table>

Note. - Object-detection rates and False Object rates for blue and red material at 0.8% of total number of pixels (np) as threshold.

According to Table 6.3, the SE-based techniques outperform the other two approaches, and the results for the blue material as for SHARE 2010, show perfect detection rates. ROC curves for SHARE 2010 and 2012 also reflect that the detection performance for the blue material is better than the detection performance for the red material. This fact can be justified by two reasons. Blue material spectra have a peak at 0.5 $\mu$m (see Figure 6.6b) which constitutes a unique and characteristic feature that could facilitate the differentiation of this material against other materials, and consequently, boosts its identification and detection. The other factor that can influence in the detection is that for the blue material, the space comprised by the field measurements of the blue target signatures encompass better the actual blue target space in the image (see blue and red material spectra in Figures 6.6b and 6.6c).
6.2.5 Target detection results for MUUFL Campus

As it was explained at the beginning of this section, this scene presents a greater detection complexity than the other two data sets. Although the original data set has ground truth information for 4 different materials, only one of those materials is considered as target material in the experiments and results described below. Therefore, the analysis is performed for the peagreen material and for the four detection techniques, SE-SSKP with $k = 20$ and SE-SSKP with Adaptive $k$, LE-ACE, and PCA-ACE. The threshold for the binary version of the detection maps are again set to the top 0.8% of the total pixels ($np = 59449$). The estimated dimensionality in each domain are: 2 for the Schroedinger embedding using $k = 20$, 3 when Adaptive $k$ ($k_{max} = 60$) is used, 3 for the Laplacian embedding and 4 for the PC embedding.

The results are shown in Figure 6.13 and they show that only two or three of the 15 target panels are identified. Those are the largest panels (3m x 3m) highlighted by the yellow squares in the binary images. As to the ROC curves, the four curves have a similar shape which means that the four techniques have a similar performance. However, there is a little deviation of SE-SSKP; $k = 20$ curve at high false alarm rates. This indicate that this SE-based method has the best rate of detection at high false alarm rates. Then, comparing the four ROC curves in the entire range of thresholds, the best detection is obtained for SE-SSKP with $k = 20$, i.e., this can be numerically supported by computing and comparing the area under the curves. The area under curve for a perfect ROC curve is 1 and its values is an indicative of the technique’s ability to distinguish the target material. For the four cases, the area under the cover are: 0.63 for SE-SSKP; $k = 20$, 0.51 for SE-SSKP;Adpk, 0.54 for LE-ACE and 0.55 for PCA-ACE.

The rates in Table 6.4 also reflect the tight numerical differences among the four techniques’ performances. The PCA-ACE and SE-SSKP with $k = 20$ show the two best detection rates, with PCA-ACE's performance the best. However, its false object-alarm rate is slightly higher than the one for SE-SSKP with $k = 20$. All these numerical and visual results show how difficult it is to detect sub pixel targets. Although the area under the curve values reflect a poor accuracy for the four techniques, the best area value is the one for SE-SSKP; $k = 20$, which indicate that this SE-based method has the best performance.
Figure 6.13: MUUFL campus detection results for peagreen material. (a) RGB rendering with ground truth. (b) ROC curves for the four detection approaches. Detection maps and binary images with the 0.8% of the top pixels as threshold: (c) SE-SSKP and $k = 20$, (d) SE-SSKP and Adaptive $k$, (e) LE-ACE, (f) PCA-ACE.
Table 6.4: MUUFL results for peagreen material

<table>
<thead>
<tr>
<th>Detection Technique</th>
<th>Peagreen material</th>
<th>( r_{OD} )</th>
<th>( r_{FOA} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>SE-SSKP; ( k = 20 )</td>
<td>0.2</td>
<td>7.82 ( \times 10^{-3} )</td>
<td></td>
</tr>
<tr>
<td>SE-SSKP; Adpk</td>
<td>0.13</td>
<td>7.93 ( \times 10^{-3} )</td>
<td></td>
</tr>
<tr>
<td>LE-ACE</td>
<td>0.06</td>
<td>7.95 ( \times 10^{-3} )</td>
<td></td>
</tr>
<tr>
<td>PCA-ACE</td>
<td>0.26</td>
<td>7.91 ( \times 10^{-3} )</td>
<td></td>
</tr>
</tbody>
</table>

Note. - Object-detection rates and False Object rates for the pea green material at 0.8\% of total number of pixels (np) as threshold.

6.3 Target detection results when the target is absent

Something interesting to analyze is how the SE-based detection technique behaves when the target material that is being sought in a given scene is not present in the scene. In this way, the blue material spectra from SHARE 2010 and 2012 is considered as the target material, and a scene from the SHARE 2012 data where the blue felt panels are not present is chosen. This SHARE 2012 scene is displayed in Figure 6.14 along with the blue material spectra and some in-scene materials spectra. The scene covers diverse materials such as gravel, concrete, soil, etc., and objects such as trees, cars, roofs.

Three of the four detection techniques - SE-SSKP; \( k = 20 \), LE-ACE and PCA-ACE - were applied to this data set. Because ground truth is not possible for this scene, only the detection maps and binary images are computed. These maps for the three techniques show bright pixels indicating that they are most likely to be identified as target. For SE-SSKP technique the material or objects identified as targets are the houses’ roofs, while for LE-ACE and PCA-ACE, trees, bushes and their shadows are identified as target. These spectra that are in Figure 6.14b show that the blue material spectrum has some features that can be similar to features of the diverse in-scene spectra, especially for the roof spectrum. Both, blue material and roof spectra have peaks at the blue visible region but displaced from each other by about 0.1\( \mu m \).
This feature can boost the spectral similarity between the blue signature and the roof spectrum which influences the graph creation that is part of the SE-base methodology. Since each pixel is always forced to be connected in the graph to a certain number of spectral neighbors, the blue target signature injected to the data is going to be connected to the most similar pixels that are present in the scene. Consequently, those neighbors will have high values in the detection map, indicating that they are the pixels most likely to belong to the target class. Basically, when a certain spectrum is injected to the data some kind of connection is forced, and if the material of interest is not present in the scene under analysis, false alarms are generated. This generation of false alarm also occurs in the other two detection techniques. However, for LE-ACE and PCA-ACE, the false alarms are tree and shadow pixels. The spectral similarity between tree/shadow spectrum and the blue signature is due to the fact that the blue panels were deployed on grass which can influence in the spectrum measurement (the blue fabric panels allow the light rays go through).

In addition, the two hypothesis involved in a target detection problem: target absent and target present could be contrasted and analyzed to see if a threshold that works well for both scenarios can be found. The idea is that when the same threshold is applied to the two scenarios, a good detection can be achieved with a minimum of false alarms (for the target present case) that is maintained for the case where the target is absent. In this way, 37 pixels from four blue target panels of SHARE 2012 scene are picked and injected to the target absent scene (Figure 6.14a) as substitutes...
Figure 6.15: Detection maps and binary images with the 0.8% of the top pixels as threshold: (a) SE-SSKP and $k = 20$, (b) LE-ACE, (c) PCA-ACE.
for some of the pixels in different areas of grass. The 37 target pixels belong to four different target panels located at different areas along the scene, panels in shadows and at open field are considered and are pointed out in Figure 6.16 by blue arrows. Figure 6.17 shows the two scenarios: target absent and target present, as well as, the detection scores that are presented as histograms when SE-SSKP with \( k = 20 \) detector is applied. For the target present scene, the pixels containing the target material are framed by magenta squares and their response to the SE-SSKP detector are represented by stems located at the detection output magnitude. The detection scores range from small positive values to the hundred thousands, although most of the background pixels have small detection magnitudes. This non-uniform background distribution makes difficult to identify how far apart the background and the target classes are, so thus a close-up of the histograms is needed.

Figure 6.16: RGB rendering of SHARE 2012 scene with magenta squares highlighting the locations of blue and red target panels, and blue arrows indicating the four blue panels injected in the target absent scene.
Figure 6.17: Target present vs. target absent hypothesis. (a) RGB rendering of a target absent scene and its response to the SE-SSKP, $k = 20$ detector represented by its histogram. (b) RGB rendering of a target present scene and its response to the SE-SSKP, $k = 20$ detector represented by the background histogram in blue and target stems in magenta.

Figure 6.18a shows the close-up histograms for the target absent scene at the left and the target present histogram at the right. Binary versions of the detection maps after applying thresholds are also shown in Figure 6.18. For the target present case, the locations of the injected target pixels are highlighted by magenta squares. The separation between the background and some target pixels is very tight. However the targets are spread in a large range of detection scores, which gives the possibility to set a threshold that gives a good detection with a few number of missed targets. This is reflected in the detection maps for the target present scenario (images at the right), where the four target panels are detected when two really different thresholds are used.
The two thresholds differ in almost ten thousand units, their values are 1500 and 10500. The images at the right that corresponds to the target present scenario are just a visual verification that the target panels are detected when the specific thresholds are applied. The images at the left show the false alarms that could be obtained, and for both thresholds, they are a few. Quantitatively, the false alarms rates at 1500 and 10500 are $1.2 \times 10^{-2}$ and $3.6 \times 10^{-3}$ respectively. This corroborates the visual findings that for both thresholds, the false alarms are really few. We can conclude that for our SE-based target detector, it is possible to set a threshold that gives a good detection performance (i.e., detection of all target objects with a low false alarm rate) if the target is present; and if the target is absent, the false alarm rate still is low. However, these results are subject to the considered image, so it would be necessary to have a well-parameterized SE-based detector if a constant-false-alarm rate (CFAR) detector is desirable.

Another interesting fact that can be noticed from the binary images at 10500 threshold is that some of the false alarms in the target absent scene, e.g., some pixels in the right-top corner roof, are not false alarms anymore when the target present scene is considered (i.e. when target pixels were injected in the non-target scene). The false alarms for both scenarios, target absent/target present, are 120 and 83 respectively. However, this trend is not preserved when the threshold value is decreased to 1500, where the respective false alarms for the target absent/target present scenarios are: 398 and 577. Figure 6.19 shows the number of false alarms as a function of the detection threshold for both hypothesis, target is present and target is absent. This behavior was expected since the SE algorithm is a non-linear manifold transformation so the detection output of a SE-based detector will also have a non-linear response along the total range of detection values. Nevertheless, the injection of actual target pixels into the scene allow us to diminish the number of false alarms at high thresholds due to the new connections between the target signature and the injected target pixels that redefine a tighter target cluster. Therefore, it is plausible to think that injecting a target cluster into the data set instead of only a few number of target signatures, as it was made for all the data sets considered in this thesis, will help in the reduction of false alarms since probably most of the connections between the target cluster points will be within the own cluster.
Figure 6.18: Detection analysis considering the two hypothesis: target is present (right column)- target is absent (left column): (a) Background and target responses to the SE-SSKP, $k = 20$ detector represented by the background histogram in blue and by the target stems in magenta. (b) Binary images using a threshold of 1500 for both hypothesis, (c) Binary images using a threshold of 10500 for both hypothesis.
6.4 Computational time and resources

This section summarizes the computational cost and resources that are involved in the development of the four detection techniques that were explained and assessed, and whose results have been presented in this Chapter. The four detection techniques and the assessment metrics were mostly implemented in MATLAB and all the experiments were run on an AMD computer with 1.4 GHz processor. The three detection techniques based on the non-linear transformations, LE and SE, have two steps in their algorithm that are the most time-consuming. These are the construction of the graph and the solution of the eigenproblem.

The graph creation step involves a searching problem, where distances for each pixel in a given scene has to be computed. In our case, the searching is solved by using the ATRIA tool which is implemented in C++, and a mex file was generated in order to use ATRIA from MATLAB. The ATRIA tool is used with the idea to reduce the time involved in the searching. As to the eigenproblem step, a MATLAB implementation of the Jacobi-Davidson iteration method (JDQR) [76] that it is effective for large and sparse matrices and is used for computing a number of selected eigenvalues and the associated eigenvectors of a matrix that can be real or complex. The MATLAB package is available online and contains the functions (M-files) that

Figure 6.19: Number of false alarms vs. detection thresholds for the two hypothesis: target is present and target is absent.
are used to solve the eigenproblem \[77\].

The experimental execution times for each technique were measured using the built-in MATLAB functions tic and toc, which together record the internal time of execution of the MATLAB lines in between. The times summarized in Table 6.5 vary from around 2 seconds for PCA-ACE technique to 25 minutes for SE-SSKP method with adaptive \( k \). As it was expected, shorter times correspond to the detection technique based on the linear transformation (PCA) and the opposite for the SE-based and LE-based methods. Also it is noted that the execution times depend on the size of each data set (The sizes of the considered images are shown in Table 6.6), and although the times for the three non-linear based techniques are comparative, the use of the adaptive estimation of \( k \) increases the runtime. Basically, the use of non-linear manifold algorithms helps to improve the detection performance but at expense of the execution times. We can afford to have these times from the academic perspective, but perhaps for more practical applications, it is desirable to accelerate the execution of these algorithms. In our approach, the most consuming time step is the construction of the spectral graph, since the searching for \( k \) is a brute-force process whose execution time depends on the size of the data sets. Therefore, it would be interesting to explore alternative ways to implement this specific algorithm. Some ideas related with this issue will be presented in the next Chapter as an avenue for future works.

Table 6.5: Execution times for the detection techniques

<table>
<thead>
<tr>
<th>Detection Technique</th>
<th>Blue Tgt</th>
<th>Red Tgt</th>
<th>Peagreen Tgt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SH10</td>
<td>SH12</td>
<td>SH10</td>
</tr>
<tr>
<td>SE-SSKP; ( k = 20 )</td>
<td>246.92</td>
<td>530.74</td>
<td>159.43</td>
</tr>
<tr>
<td>SE-SSKP; Adpk</td>
<td>249.55</td>
<td>1467.8</td>
<td>229.67</td>
</tr>
<tr>
<td>LE-ACE</td>
<td>233.75</td>
<td>615.13</td>
<td>184.48</td>
</tr>
<tr>
<td>PCA-ACE</td>
<td>2.05</td>
<td>5.56</td>
<td>2.25</td>
</tr>
</tbody>
</table>

SH10 and SH12 are abbreviations for SHARE 2010 and SHARE 2012. Tgt is the abbreviation for target material and the times are reported in seconds.
Table 6.6: Size of the processed images

<table>
<thead>
<tr>
<th>Image Size</th>
<th>SHARE 2010</th>
<th>SHARE 2012</th>
<th>MUUFL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12264</td>
<td>47600</td>
<td>59449</td>
</tr>
</tbody>
</table>

Image sizes are reported in pixels.

### 6.5 Summary

Detection results for hyperspectral data sets were presented in this Chapter. These results include TAD detection results as an initial exploration of the applicability of graph-based techniques in target detection, and also the SE-based techniques’ results that show the potential of the Schroedinger algorithm in the exploitation of hyperspectral images. The SE-based methodology using a global $k$ and its variation where the parameter $k$ is adaptively estimated were compared against two other methods that use the alternative non-linear manifold method LE, and the linear transformation PCA. In these two methodologies, LE and PCA are only used as transformation techniques and the actual detection is performed by using the widely used ACE detector. The four techniques were applied to three different data sets that contain various target panels deployed around distinctive regions with diverse detection complexity. The SE-based method, SE-SSKP; $k = 20$ mostly outperforms the other three techniques, even the SE-based method with the adaptive estimation of $k$. However, for the sub pixel detection experiment, the four techniques have similar fair performance showing the level of difficulty in the sub-pixel detection problem. The next Chapter will summarize the findings and contributions of this research work, as well as, the suggestions for future work.
Chapter 7
Conclusions

This thesis has been focused on the exploration and development of a spectral target detection technique based on non-linear manifold methods. This technique is based on the Schroedinger Eigenmaps (SE) algorithm and uses a particular spatial-spectral graph for improving the identification and detection of target pixels. The aim was to develop a completely novel target detection approach based on the local structure of the data, represented by a graph, instead of the traditional detection approaches that use statistical or geometric models. Therefore, the SE-based method developed in this work, does not require any statistical or geometric assumptions about the data and since the Schroedinger embedding can be steered in convenient directions, the detection operator itself does not require the use of any of the traditional detectors described in Chapter 2. The proposed technique is completely based on the graph data structure and includes a complete framework covering all the steps from the graph data representation to the actual detector definition. It offers an alternative way to look at the data when a target detection issue is being addressed. This Chapter presents a summary of the contributions of this thesis and some suggestions for future research work.

7.1 Contributions of this work

7.1.1 Adaptive Estimation of Parameters

The SE algorithm described in Chapter 5 includes steps where various parameters must be set or estimated. The parameter of scaling $\sigma$ in the spectral weighting function, the $k$ number in the nearest neighbor searching, and the $\hat{\beta}$ in the barrier potential definition are some of the parameters of interest. The contribution of
the thesis related with this topic is the introduction of adaptive approaches for estimating $\sigma$ and $k$ in the graph creation step, and the design of a experiment for tuning the $\hat{\beta}$ parameter involved in the definition of the Schroedinger operator.

Parameters in the Graph creation

$k$ and $\sigma$ parameters involved in the graph creation step are in most of the cases set to a unique and global value. These values are manually picked based on the expertise of the user or by trial and error. Since this target detection technique is based on the graph data structure, the challenge was to set an appropriate $k$ value that represent as best as possible the local structure of the data. Therefore, the idea was to set various $k$ values that suit different pixels, and whose estimation is adaptively performed such that those pixels in high density that have numerous neighbors will be assigned a large $k$, and those pixels in low density that have a few neighbors will be assigned a small $k$. The other parameter $\sigma$ is a scaling factor in the weighted matrix definition that determines when two vertices can be considered similar. The weighted matrix that was described in Section 3.2.1 represents the connectivity of the graph and how strong the connection is between any two vertices. In general, $\sigma$ is empirically set to a single value for all pixels in the data set, but in many cases, one value may not work well for all pixels in an image since the data sets have multiple local distributions. Then, as for $k$, an adaptive estimation of $\sigma$ was introduced with the purpose of better capturing the local structure of the data set. The estimation depends on the local distribution and generates high weights for edges within clusters and low weights for edges across the clusters.

These adaptive estimation approaches were introduced with the hope of getting a more faithful representation of the local data structure, where relevant connections between data points are preserved and not get cluttered by over-connections or missed by under-connections imposed by adverse $k$ or $\sigma$ values. The SE-based detection technique using the adaptive $k$ showed to have a comparative performance with the one using $k = 20$, which in most of the experiments had the best performance. This fact shows that this adaptive estimation is an alternate way for creating the graph that offers a good detection performance.

Parameters in the Schroedinger operator

As it was described in Section 5.1, the $\alpha$ parameter that later was labeled as $\hat{\beta}$, influences the Schroedinger embedding such that labels in the potential matrix have
more impact over the connections given by the Laplace operator. After some experiments shown in Section 5.1, it was concluded that \( \hat{\beta} \) has to be large enough to get as many labeled pixels as possible around zero, but not so large that the embedding is compromised. For \( \hat{\beta} \) and for the other parameters, the estimation of \( \hat{\beta} \) becomes a trade-off problem, where a very little analysis to find a way to pick an appropriate \( \hat{\beta} \) has been done. In this way, an experiment aimed at finding an appropriate value for \( \hat{\beta} \) was designed and introduced in this thesis. In the experiment, \( \hat{\beta} \) was varied within a range of values and for each one, the detection map was computed as a function of the number of Schroedinger Eigenmaps. Then, the appropriate \( \hat{\beta} \) was chosen as the value where a maximum number of Schroedinger Eigenmaps can be used in the detection with a minimum rate of false alarms.

This experiment for estimating \( \hat{\beta} \) is first, an attempt to reduce the number of parameters that have to be manually set, and second, is a more structured way to estimate an important parameter that directly influences the Schroedinger embedding and the detection performance.

### 7.1.2 Knowledge propagation scheme and spatial integration into SE approach

According to what was described in Section 5.2, the barrier potentials generated by small sets of labeled pixels impacts in a “brittle” manner the propagation of those constraints to nearby points in high dimensions, and in turn, affects the detection performance. In this fashion, the “knowledge propagation” scheme was introduced with the idea to add connections to the barrier potentials, and through them propagate these constraints to more nearby pixels. This propagation approach also gave us the opportunity to integrate the spatial information to the SE-based methodology that so far was only based on the spectral connectivity. The idea behind the use of spatial connections was to reinforce weak spectral connections between pixels that spatially are closer neighbors. The effectivity of this knowledge propagation scheme was shown by an experiment using toy data. The impact of using the modified barrier potential matrix in the Schroedinger embedding was shown and compare against the use of the typical barrier potentials.

Since the proposed target detection methodology uses an external target spectrum as prior information, i.e., a target spectral signature measured at the field, the definition of the potential matrix and its modified version (modified by the knowledge propagation scheme) can be seen as a learning process, where data points that
are spectral and spatial neighbors of the target material are identified from the graph data structure, and are considered for defining the modified barrier potential matrix. This learning process and the integration of knowledge propagation scheme to the Schroedinger algorithm allows us to capture and consider additional features of the data in order to improve the separability between most of the target pixels and the background. This learning process and the propagation of the learned constraints are considered as an important contribution of this thesis, since they, as a new modification to the original Schroedinger Eigenmaps algorithm, have helped to improve the detection of target pixels.

7.1.3 Detection operator

The detection operator is the key tool in any target detection methodology where the detectors are derived from competitive hypotheses that model the two scenarios: target is present or target is absent. Classical detectors use statistical and geometric models to define the detection operators. In this way, most of the detection techniques based on the non-linear manifold learning use the manifold technique as a transformation method and then the classical approaches, i.e., statistical and geometric models, are used as detectors. On the contrary, in the SE-based target detection methodology, there are two key elements. The first one is the Schroedinger operator, that is the responsible to embed the original hyperspectral image into the new space. The second element is related to the detection itself, where the transformation into the detection space is performed taking advantage of a nice property of this particular embedding. This property helps to enhance the separation between a particular material vs. the rest of materials present in the data set. This fact allows us to use in a different way a non-linear manifold algorithm in a detection approach, and it is one of the most important contributions of this thesis. In Chapter 5 a detector was proposed based on the graph data structure that exploits the theory behind the Schroedinger embedding, which states that the lower dimensional representation of the data is steered in certain directions lead by the barrier potential. In this way, the barrier potentials and their nearby pixels, i.e., target-like pixels, are pulled toward the origin and then, the proposed detector takes advantage of the this fact and assigns higher values to points close to the origin. This assignation is basically a computation that considers the inverse of the magnitude of each pixel in the Schroedinger embedding. Its definition and a toy example was shown in Section 5.3.2 and its usefulness was tested by applying the entire SE-based method to three different hyperspectral data sets.
7.1.4 TAD applied to target detection

In Chapter 3, a graph-based method applied to anomaly detection was discussed. This is the Topological Anomaly Detection (TAD) algorithm that is purely based on a graph data structure. The original TAD uses the graph on the data to model the background points and then computes an anomalousness score for each pixel based on a co-density measurement, i.e., sum of distances to some $k$ nearest neighbor. This original implementation was applied to hyperspectral imagery and showed improved results over more traditional anomaly detectors. A later study shown the robustness of the background model created by TAD, when different ways to create a graph were tested. Based on this fact, an analysis of the background components created by TAD applied to target detection was performed as an initial step of the framework of this thesis.

In this study, the background pixels estimated by TAD were characterized in order to get background basis or background statistical parameters and used them in geometric or statistical detectors. The results of this initial work showed the potential applicability of graph-based methods in target detection. The fact that TAD had a good performance in HSI tasks beyond anomaly detection, encouraged us to continue studying the potential and applicability of another graph-based methods in target detection. Specifically, we focused on the non-linear manifold learning algorithms that use a graph as a representation of the local structure of the data.

7.2 Future Work

As it was described previously, the main objective and contribution of this thesis was to develop a target detection technique completely based on a non-linear manifold learning algorithm, which uses a graph model of the data instead of the classical statistical or geometric models - that are widely used by the typical target detectors. The motivation behind this research was the limitations imposed by the linear or parametric models that are assumed by the classical detection methods, especially when the spatial and spectral resolution of the hyperspectral sensors is increased progressively and such amount of data can clutter relevant features of the hyperspectral data. Continuing in this vein, the proposed SE-based technique has shown to have in general a good detection performance even when it was compared against one of the most widely used detectors - the ACE detector. At least for the cases presented in this thesis, the SE-based methods outperformed the other two detection techniques based on the Laplacian Eigenmaps algorithm and on the linear
transformation method, PCA. This does not mean that graph-based or non-linear based approaches are always better than the other existing detection approaches, but they offer a different way of modeling the data and designing task-specific algorithms.

One of the detection problems that was considered in Chapter 6 was the detection of sub pixel targets, i.e., the target material only comprises part of the pixel under observance. Results showed that the four considered techniques, even the ones based on the SE algorithm, have a poor/fair performance. This issue could be addressed in a future research work, where actual in-scene background and target materials are used to model a target cluster, that can be injected into the data as the same way as the proposed SE-based method does. In our approach, we have used a kind of target cluster formed by the available field measurements of the target material spectra. However, looking at the target spectra and comparing the in-scene spectra against the field measurement (see Figures 6.6b, 6.6c and 6.7b), there are some small differences in shape at specific wavelength regions, especially for the MUUFL campus data (the sub pixel target detection), which influences the detection results.

Another avenue of investigation is the exploration of adaptive methods for estimating the parameter $k$. According to what was shown in this thesis, the use of various $k$, whose estimation is based on the local density of the region where each pixel is, impacts the graph representation (see Figure 5.8), the Schroedinger embedding and in turn, the detection performance. The detection results for SE-based technique when the adaptive $k$ is used, exhibits a good performance very comparative with the SE-based method with $k = 20$, which is the approach with the best detection performance. Since $k = 20$ is not a magic number that can fail in the data structure representation of other images, it is interesting to carry out a more thorough study of different methods for estimating $k$ that could be evaluated from different perspectives: (1) From a data representation point of view the distribution of edges or number of connected components can be analyzed and compared with the natural data distribution and the actual classes present in the scene. (2) From the detection performance point of view. As it has been shown in this thesis, the graph data representation directly influences in the Schroedinger embedding and in the detection performance, so each of the possible techniques for estimating $k$ could be quantitatively assessed by using detection metrics such as ROC curve or FAr.

Additionally, in Section 6.4 was shown that the SE-based methodologies are the
most computational expensive algorithms compared against the use of PCA and LE. Furthermore, the use of the non-linear manifold algorithms such as LE and SE increased the processing time of the different HSI data sets considered in this thesis. Therefore, it would be convenient to explore methods or ways for improving the processing times. It could be important to focus on the most expensive steps in the SE-based detection methodology such as the construction of the similarity graph. In this way, two schemes, “Laplacian Eigenmaps with Random Projections (LERP)” and a “Fast Approximate Neighborhoods” introduced by Halevy in his PhD dissertation could be considered [78]. According with the thesis, random projection scheme can be used as a pre-processing step to map the original data to a low-dimensional space where the LE algorithm is applied. The dimensionality reduction introduced by the random projections scheme reduces the computational time but preserves the output of LE as if it had been directly applied to the data. Regarding the Fast Approximate Neighborhoods, the time reduction is rather focused on the searching for nearest neighbors, where the searching that originally is performed on each and every data point (which implies impractical times for large data sets) is now performed recursively on subsets of data points; and the final solution is the set of subset solutions such that if a data point belongs to different subsets, its nearest neighbors will be selected from the nearest neighbors found in each subset.
Bibliography


[42] T. L. Ainsworth, C. M. Bachmann, and R. A. Fusina, “Local intrinsic dimensionality of hyperspectral imagery from non-linear manifold coordinates,” in


Bibliography


[75] W. Czaja. Personal communication (July 7th, 2016).

