SUPERPIXEL-BASED TARGET DETECTION METHODS FOR HYPERSPECTRAL IMAGES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MUSTAFA KÜTÜK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2018

Approval of the thesis:

SUPERPIXEL-BASED TARGET DETECTION METHODS FOR HYPERSPECTRAL IMAGES

submitted by **MUSTAFA KÜTÜK** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Tolga Çiloğlu
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. A. Aydın Alatan
Supervisor, **Electrical and Electronics Eng. Dept., METU**

Dr. Alper Koz
Co-supervisor, **Center for Image Analysis (OGAM), METU**

Examining Committee Members:

Prof. Dr. Buyurman Baykal
Electrical and Electronics Eng. Dept., METU

Prof. Dr. A. Aydın Alatan
Electrical and Electronics Eng. Dept., METU

Assoc. Prof. Dr. Uğur Murat Leloğlu
Geodetic and Geographical Information Tech. Dept., METU

Assist. Prof. Dr. Sevinç Figen Öktem
Electrical and Electronics Eng. Dept., METU

Assist. Prof. Dr. Seniha Esen Yüksel
Electrical and Electronics Eng. Dept., Hacettepe University

Date: **06.09.2018**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Mustafa Kütük

Signature :

ABSTRACT

SUPERPIXEL-BASED TARGET DETECTION METHODS FOR HYPERSPECTRAL IMAGES

Kütük, Mustafa
M.S., Department of Electrical and Electronics Engineering
Supervisor : Prof. Dr. A. Aydın Alatan
Co-Supervisor : Dr. Alper Koz

September 2018, 125 pages

Spectral signature-based methods which form the mainstream in hyperspectral target detection can be classified mainly into three categories. These are the background modeling methods, subspace projection based methods, and hybrid methods that combine linear unmixing with abundance estimation. A common characteristic of all these methods is to classify each pixel of the hyperspectral image as a target or background while ignoring the spatial relations between neighboring pixels. Integration of contextual information defined over neighboring pixels can, however, suppress the noise on the individual pixels and yield better detection. In this thesis study, the baseline superpixel extraction algorithms which are previously developed for RGB images, namely the Simple Linear Iterative Clustering (SLIC) algorithm and boundary update-based superpixel extraction method, are first adapted to hyperspectral images. Then their extraction performances are compared in terms of the metrics which are boundary recall and undersegmentation error. After the selection of the boundary update-based superpixel extraction algorithm due to its better performance, different target detection methods performing over superpixels are proposed. The proposed methods utilize superpixel representatives instead of pixels for background modeling, matching and abundance estimation. The experiments suggest that using superpixels for target detection improves the detection performances in terms of precision-recall curves compared to the baseline methods using only pixels.

Keywords: Superpixel, Hyperspectral Images, Target Detection, Hybrid Detectors, Linear Unmixing

ÖZ

HİPERSPEKTRAL GÖRÜNTÜLER İÇİN SÜPERPİKSEL TABANLI HEDEF TESPİT YÖNTEMLERİ

Kütük, Mustafa
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. A. Aydın Alatan
Ortak Tez Yöneticisi : Dr. Alper Koz

Eylül 2018 , 125 sayfa

Hiperspektral hedef tespiti içinde ana akımı oluşturan spektral imzaya dayalı yöntemler temel olarak üç kategoriye ayrılabilir. Bunlar, arka plan modelleme yöntemleri, altuzay izdüşümüne dayalı yöntemler ve lineer karıştırmayı bolluk tahminiyle birleştiren hibrit metotlardır. Tüm bu yöntemlerin ortak bir özelliği, komşu pikseller arasındaki mekansal ilişkileri göz ardı ederken, hiperspektral görüntünün her bir pikselini bir hedef veya arka plan olarak sınıflandırmaktır. Bununla birlikte, komşu pikseller üzerinde tanımlanan bağlamsal bilgilerin entegrasyonu, tek tek pikseller üzerindeki gürültüyü bastırabilir ve daha iyi saptama sağlayabilir. Bu tez çalışmasında, RGB görüntüler için geliştirilen süperpiksel algoritmaları olan Basit Doğrusal Tekrarlamalı Kümeleme (SLIC) algoritması ve sınır güncellemeye dayalı süperpiksel çıkarma yöntemi hiperspektral görüntülere uyarlanmıştır. Daha sonra onların süperpiksel çıkarma performansları sınır geri çağırma ve alt bölümleme hatası metrikleri kullanılarak karşılaştırılmıştır. Daha iyi performans göstermesi nedeniyle sınır güncelleme tabanlı süperpiksel çıkarma algoritmasının seçilmesinden sonra, süperpikseller üzerinden gerçekleştirilen farklı hedef tespit yöntemleri önerilmiştir. Önerilen yöntemler, arka plan modelleme, eşleştirme ve bolluk tahmini için piksel yerine süperpiksel temsilcileri kullanmaktadır. Deneyler hedef tespiti için süperpiksel kullanımının yalnızca piksel kullanan temel yöntemlerle doğruluk-geri çağırma eğrileri açısından karşılaştırıldığında tespit performansını geliştirdiğini ortaya çıkarmıştır.

Anahtar Kelimeler: Süperpiksel, Hiperspektral Görüntüler, Hedef Tespiti, Hibrid Hedef Tespit Algoritmaları, Doğrusal Ayrıştırma

To my parents and my lovely wife Elif...

ACKNOWLEDGMENTS

I would like to thank my supervisor Prof. Dr. A. Aydın Alatan for his guidance, understanding, encouragement and support which helped me a lot to reach this level throughout this thesis work. I am grateful to him for giving the opportunity to work in Center for Image Analysis (OGAM). He is a unique person who brings young and enthusiastic people together to work for supreme purposes in this center.

I would like to acknowledge my co-supervisor Dr. Alper Koz who gave valuable suggestions and creative ideas to improve this work. Whenever I need his advice, he was always there for me. I would also like to thank for his patience and understanding.

I would like to thank Mr. İzzet Şirin and Mr. Azmi Utfan Cinek in Havelsan Teknoloji Radar Corporation for giving the opportunity to work in this research.

I am grateful to Bertan Günyel for his support and advice. He is one of the most talented engineers I have ever met. I made crucial changes in my working style by being influenced by his ideas.

I would like to especially thank my colleagues İzlen Geneci, Okan Bilge Özdemir, and Kerem Tütüncüoğlu for their huge contributions to obtain data.

I wish to express my sincere thanks to my wife Elif Sarıgöz Kütük for her endless support and confidence to me. She always believes that I will be successful, therefore I am self-confident and motivated. She encouraged me to finish this study.

Finally, I would like to thank my parents and my brother for their support, understanding, and encouragement. They always believed that I will be a beneficial person to the community. I hope that I will realize their wishes.

TABLE OF CONTENTS

CHAPTERS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

SWIR	Short-Wavelength Infrared
GLRT	Generalized Likelihood Ratio Test
SAM	Spectral Angle Mapper
ACE	Adaptive Coherence Estimation
DTDCA	Desired Target Detection and Classification Algorithm
SMF	Spectral Matched Filter
OSP	Orthogonal Subspace Projection
CEM	Constrained Energy Minimization
TCIMF	Target-Constrained Interference-Minimized Filter
OPCI	Orthogonal Projection Correlation Index
ROC	Receiver Operating Characteristic
FCLS	Fully-Constrained Least Squares
NNLS	Non-Negativity Constrained Least Squares
AMSD	Adaptive Matched Subspace Detector
HSD	Hybrid Structured Detector
HUD	Hybrid Unstructured Detector
SLIC	Simple Linear Iterative Clustering
MSE	Minimum Square Error
SID	Spectral Information Divergence
LCTF	Liquid Crystal Tunable Filter
SP	Superpixel
BU	Boundary Update Based Superpixel Extraction Method

CHAPTER 1

INTRODUCTION

Hyperspectral target detection is a growing research field in areas such as remote sensing, mining, defense industry, biomedical, optics, forensic applications, food safety and chemistry [4–7]. Detection of an enemy vehicle from an image which is obtained by using satellite, finding a region which includes minerals or detection of a rotten food at the product band of a food factory can be given as example situations for hyperspectral target detection.

Hyperspectral images include more bands than RGB images in the electromagnetic spectrum. Detection of target materials by analyzing spectral signature of the target material in infrared bands is superior to analyzing only visible bands due to the existing characteristic information in the infrared range.

Many different approaches have been developed for target detection over the years. A general scheme for the target detection algorithms is given in Figure 1.1. These approaches are classified in three main groups as anomaly detectors, unmixing algorithms, and signature-based target detection algorithms.

Anomaly detectors separate pixels which have different spectral characteristics from background without using any supervised spectral signatures. Anomaly detectors can be used when there is no information about the spectral signature of the targets. However, if other sparse materials exist in the scene, then the resultant anomalies could be different than the target material. For this reason, anomaly detection can be thought as a basic operation for target detection. Reed-Xiaoli (RX) anomaly detector, which is the most popular anomaly detector, models the background as a multivariate normal distribution whose statistical parameters such as mean and covariance are estimated by using pixels in the image [8, 9]. The simplest definition of RX anomaly detector is the calculation of the square of the Mahalanobis distance between the test pixel and the local background mean.

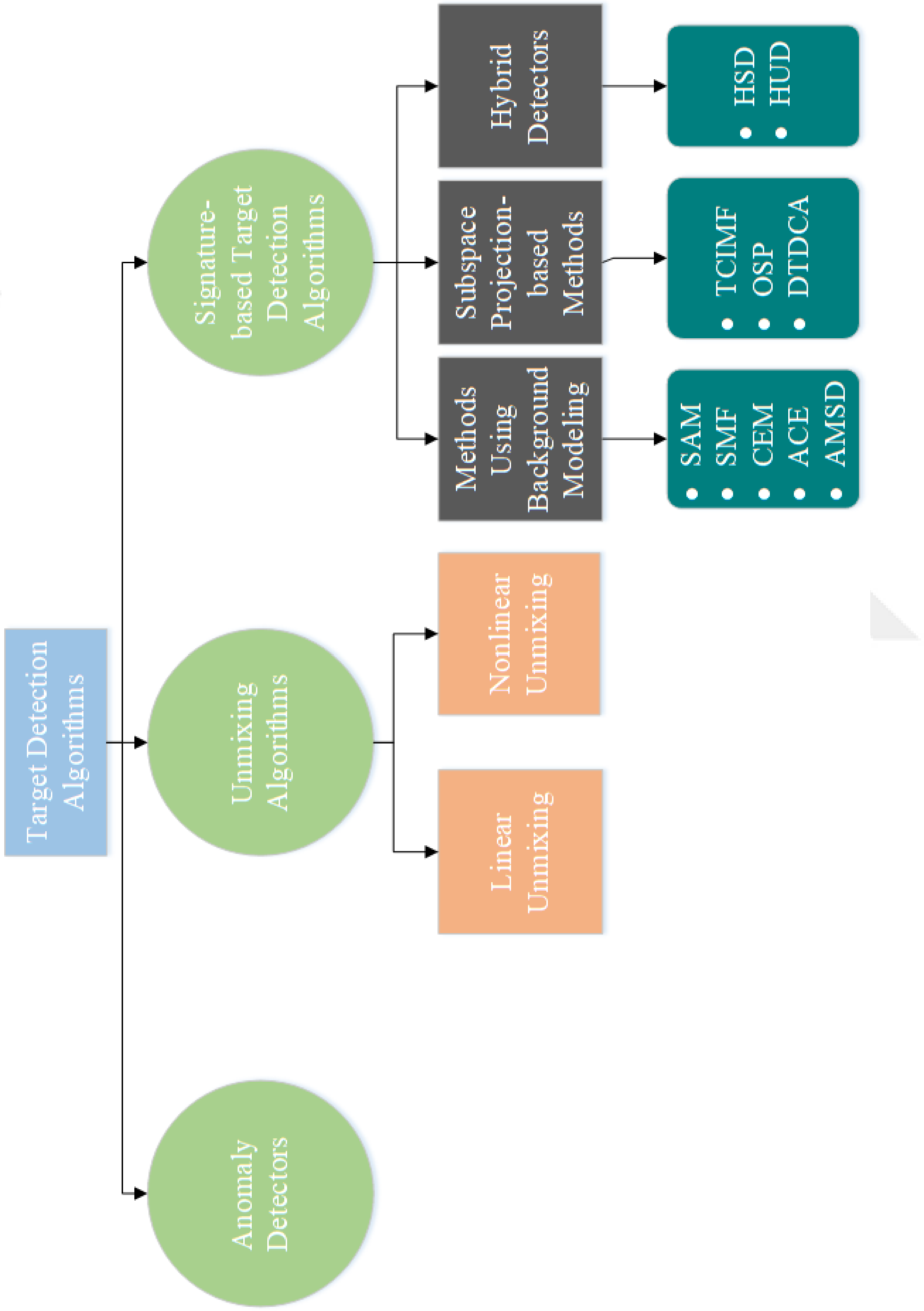

Figure 1.1: The general scheme of the target detection algorithms

Unmixing algorithms assume that a pixel can be mixture of different endmembers such as target material and background. The aim of the unmixing algorithms is to accurately obtain the ratios of target and background materials in these mixed pixels. Detection of pure pixels consisting of only one endmember is a well treated problem which is simply solved with spectral signature-based target detection algorithms. However, when there are mixed pixels consisting of the target material and background in the scene, unmixing algorithms are more successful than other types of target detection algorithms.

There are two main classes for unmixing algorithms: linear unmixing and nonlinear unmixing. Linear unmixing, as it can be understood from its name, assumes that the observed spectrum is a linear combination of the spectral signatures of endmembers. Unlike the linear unmixing, nonlinear unmixing assumes that there is also nonlinear effect such as interactions between endmembers in the reflected light which reaches to the sensor [5, 10].

As the last category, signature-based target detection algorithms take hyperspectral image and spectral signature of the target as inputs and calculate the distance or similarity between the spectral signature of the target and pixels, which are formed of vectors containing the spectral data of the hyperspectral image. There are many different approaches for the signature-based target detection in the literature [8, 11]. Most of these methods use background covariance matrix which is usually computed from all pixels of the image to detect target material. Some of them also use spectral signatures of non-target (background) materials as inputs. Taking into account this information, it is possible to separate background and target materials by using signature-based target detection algorithms.

Another approach of the spectral signature-based target detection algorithms is hybrid detectors which combines the signature-based methods and linear unmixing. After abundance map of the target material is obtained with the unmixing methods, these abundance maps corresponding to target and background materials are multiplied with spectral signature of endmembers and the result is the input of hybrid detectors. In this way, hybrid detectors benefit from advantages of both linear unmixing and signature-based target detection methods [12].

The main idea behind these algorithms is to analyze the hyperspectral image without taking spatial relations between the pixels into account. Besides the spectral data, exploiting the neighboring relations between pixels as spatial information can increase the detection performance for the cases when the target covers more than one pixels. However, a challenge for such a case is to determine the exact area which includes the target pixels. The utilization of superpixels, which use the spatial and spectral information to group similar pixels, can be considered as a novel solution which can improve the accuracy of the system for this problem. In such a solution, extracting and using superpixel representatives, instead of individual pixels, provides the possibility of faster and more efficient analyses for classification as there are fewer superpixel representatives than the number of pixels of the image. A statistical analysis performed over the locally similar parts obtained with superpixels is superior to an individual analysis performed over all image pixels due to reducing of noise effects [13].

There are different approaches for obtaining superpixels of an RGB image in the literature [14–16]. However, the number of superpixel algorithms developed for hyperspectral images are limited. Most of these studies mainly adapt a developed method for RGB images to the hyperspectral images. For instance, the adaptation of the commonly known SLIC algorithm is performed by Psalta et al. [17]. Similarly, Caliskan et al. [18] also adapts the boundary update based superpixel extraction method [16], which is developed for RGB images, to the hyperspectral images. However, a basic disadvantage of these prior studies is the lack of complete and detailed experiments and comparisons.

1.1 Scope and Contributions of the Thesis

The main contribution of this thesis is first to provide detailed comparisons between different superpixel extraction methods developed for hyperspectral images and to reveal the superior one to be used for target detection. As a second contribution, superpixel-based target detection methods are proposed by adapting the pixel-wise conventional target detection methods to the extracted superpixels. In parallel with these purposes, the preparation of the related experimental setup to practice these methods in a close distance scenario can be regarded as another contribution of this research. These contributions are given in detail as follows:

- The superpixel algorithms developed for hyperspectral images are experimentally compared in terms of their performances evaluated with boundary recall and undersegmentation error metrics. While boundary recall compares the differences in boundaries of the objects in the result and ground truth, undersegmentation error calculates the rate of the non-overlapping parts between the segmented objects of the result and ground truth. The main parameters for these comparisons are selected as the number of superpixels, weights of spectral and spatial distance metrics, and the types of spectral distance metrics which are used in superpixel extraction algorithms.

- The second main contribution of this thesis is to propose superpixel-based target detection methods for hyperspectral images. The proposed algorithms consider different variations of superpixel representatives to be used instead of pixels in conventional target detection algorithms. The comparisons of the proposed superpixel-based method with the existing baseline methods are performed in terms of precision and recall curves. In this context, the three main classes of spectral signature-based target detection algorithms, which can be labeled as the methods using background modeling, subspace projection based methods, and hybrid methods, are studied for the development of superpixel-based detection methods. Spectral Matched Filter (SMF), Adaptive Coherence Estimator (ACE), and Constrained Energy Minimization (CEM) algorithms, which are the elements of the first group, model the background with normal distribution then apply likelihood ratio test for detection. Subspace projection based methods such as Orthogonal Subspace Projection (OSP) and Desired Target Detection and Classification Algorithm (DTDCA) use the orthogonal projection of background to discriminate target from the background. Hybrid detectors combine linear unmixing methods with signature-based methods to detect the target material. Hybrid Unstructured Detector (HUD) and Hybrid Structured Detector (HSD) algorithms belong to the hybrid detectors class. Among these algorithms, ACE, DTDCA, and HUD are selected as members of three main classes to develop superpixel-based target detection methods. Then the performances of the proposed superpixel-based target detection methods are compared with their conventional pixel-wise versions.

- As the last contribution of this thesis, new experimental setups are prepared for different types of chemical substances such as soluble and solid forms on different backgrounds by considering the scarcity of experimental datasets for close-range hyperspectral target detection. The experiments are designed to analyze various aspects such as light conditions, background types, form of the chemical substances, residue types. Although there are several datasets for hyperspectral images available on the internet [1, 19, 20], these images are not suitable for spectral signature-based target detection methods due to the lack of the spectral signatures captured from the ground.

1.2 Outline of the Thesis

The outline of the thesis is given as follows:

In **Chapter 2**, signature-based target detection methods, linear unmixing methods and hybrid target detection methods which are applied to hyperspectral images, and their differences are explained in detail.

In **Chapter 3**, Simple Linear Iterative Clustering (SLIC) and Superpixel Extraction Method based on Boundary Update methods are explained and their theoretical background is given.

In **Chapter 4**, the proposed superpixel-based target detection methods, which provide the adaptation of the superpixels to conventional target detection algorithms, are presented. Different variations of this adaptation, which can be obtained by using superpixel representatives or image pixels in every stage of the system such as calculation of the background covariance matrix, spectral signature matching and estimation of abundances, are considered and discussed for superpixel-based target detection.

In **Chapter 5**, the instruments used for the experimental setups are introduced, the datasets are given, the utilized radiance/reflectance conversion is clarified, and the performance metrics for both target detection and superpixel extraction are explained.

In **Chapter 6**, the performances of the conventional and proposed superpixel-based target methods are compared in terms of boundary recall and undersegmentation error. In addition, the target detection results for superpixel-based methods and conventional methods including background modeling, subspace projection, and hybrid methods are compared with respect to precision-recall curves.

In **Chapter 7**, summary of the thesis and conclusions are given.

CHAPTER 2

TARGET DETECTION METHODS FOR HYPERSPECTRAL IMAGERY

Hyperspectral target detection has been actively studied over the years. Different approaches to the detection problem have been developed for improving the detection performances. The spectral signature of the target should have known for the application of signature-based target detection methods. Although accurate measurement of the target spectra by using a spectrometer is possible, the sensed spectra of the same materials by a hyperspectral camera are not the same due to some variations in atmospheric conditions, material composition, location, noise of the sensor, surrounding materials, and other factors [21]. Therefore, the spectra corresponding to the pixels of target materials may include spectral variability which negatively affects the capturing of the same spectra measured by the spectrometer.

Another challenging factor to obtain the spectral signatures of the target materials is interference, which causes to spectral variability on the image pixels, between the target material and background surface. This phenomenon is called as mixing problem. Depending on the spatial resolution of the sensor, a hyperspectral image consists of pure and mixed pixels. While pure pixels contain only one surface material, mixed pixels include more than one material.

The main focus points of target detection algorithms are to overcome both spectral variability and mixed-pixel interference problems. In this chapter, different approaches of the spectral signature-based methods which are proposed to address these problems are presented and the theory behind them is explained in detail from basic idea to complex algorithms.

2.1 General Overview of Signature-Based Target Detection Methods

Target detection algorithms model the background by using some statistical information. Most of the detectors are based on the likelihood ratio test which is a useful tool giving the statistical result for a test pixel whether it is likely to be target or background by using the estimations of mean and covariance from all image [22]. Likelihood ratio test has main advantages such as minimizing the risk of wrong classification and increasing the performance by maximizing the separation between target and background spectra [21]. The likelihood ratio test is formulated by

$$\Lambda(x) = \frac{f_1(x|target\, present)}{f_0(x|target\, absent)} = \frac{f_1(x|H_1)}{f_0(x|H_0)} \tag{2.1}$$

where the conditional probability density functions of observing the test pixel spectrum x under the target absent (H_0) and target present (H_1) hypotheses are shown as $f_0(x|H_0)$ and $f_1(x|H_1)$ respectively. If $\Lambda(x)$ is greater than the predetermined certain threshold value for a given x, then the target present hypothesis is applicable. Otherwise, the target absent hypothesis is accepted as true. The corresponding conditional probability density function for each hypothesis is modeled as a Gaussian distribution. Generally, these hypotheses can be represented by

$$\begin{aligned} H_0 &: x \sim N(\mu_{\mathrm{b}}, \Sigma_{\mathrm{b}}), \\ H_1 &: x \sim N(\mu_{\mathrm{t}}, \Sigma_{\mathrm{t}}) \end{aligned} \tag{2.2}$$

where μ_{b}, μ_{t} are mean vectors, $\varSigma_{\mathrm{b}}$, $\varSigma_{\mathrm{t}}$ are covariance matrices of background and target respectively and *N* represents the Gaussian distribution. Detection algorithms are derived from these general formulations shown in Equation 2.2.

There are a lot of different methods for signature-based target detection. In this section, conventional spectral signature-based target detection methods are presented beginning from the simplest algorithm.

2.2 Methods Using Background Modeling

The background of the scene is modeled by a Gaussian distribution in these methods. Then the correlation of the pixel spectrum with the target spectrum is analyzed. As a result, a score image that shows the target and non-target pixels is obtained. The theories behind these algorithms are explained clearly in this section.

2.2.1 Spectral Angle Mapper (SAM)

One of the fundamental approaches is Spectral Angle Mapper (SAM) which calculates the angle between the target spectrum and the pixel vector in the spectral space [11]. SAM is the easiest and fastest method to observe the similarity between the target and pixels. The hypotheses for SAM can be represented by

$$\begin{aligned} H_0 &: x = b \sim N(0, \sigma^2 I), \\ H_1 &: x = \alpha s + b \sim N(\alpha s, \sigma^2 I) \end{aligned} \tag{2.3}$$

where *b* is the background noise, α represents the weight of the known target spectrum s. Background is modeled as zero-mean, white Gaussian noise with unknown variance parameter σ^2 in this method. The variances of both distributions for two hypotheses are assumed to be equal. Also the weight parameter α is another unknown parameter. These unknown parameters must be estimated to obtain the formulation of the detector which equals to Generalized Likelihood Ratio Test (GLRT).

To estimate α and σ^2 parameters, Maximum Likelihood Estimation (MLE) is used [23]. The scalar unknown parameter θ, which maximizes the probability distribution function $p(x;\theta)$ for fixed value of *x*, is defined as MLE [24]. MLE is the most practical estimator which can be used when the large data are available. MLE of the corresponding unknown parameters can be found by taking the derivative of log-likelihood functions of both hypotheses and equating them to zero. Probability distribution functions of both hypotheses for SAM can be given as

$$p(x|H_0) = \frac{1}{(2\pi)^{\frac{L}{2}} |\sigma^2 I|^{\frac{1}{2}}} exp\left\{ -\frac{1}{2} x^T (\sigma^2 I)^{-1} x \right\}, \tag{2.4}$$

$$p(x|H_1) = \frac{1}{(2\pi)^{\frac{L}{2}} |\sigma^2 I|^{\frac{1}{2}}} exp\left\{ -\frac{1}{2} (x - \alpha s)^T (\sigma^2 I)^{-1} (x - \alpha s) \right\}, \tag{2.5}$$

where L represents the size of x vector. Estimated variance parameter σ^2 can be found by using Equation 2.4:

$$ln(p(x|H_0)) = -\frac{L}{2} ln(2\pi) - \frac{1}{2} ln(|\sigma^2 I|) - \frac{1}{2} x^T (\sigma^2 I)^{-1} x\,, \tag{2.6}$$

$$\frac{\partial ln(p(x|H_0))}{\partial \sigma^2} = -\frac{L}{2\hat{\sigma}^2} + \frac{x^T x}{2\hat{\sigma}^4} = 0\,, \tag{2.7}$$

$$\hat{\sigma}^2 = \frac{x^T x}{L}\,. \tag{2.8}$$

Likewise, MLE of the weight of the target α can be obtained from Equation 2.5:

$$ln(p(x|H_1)) = -\frac{L}{2}ln(2\pi) - \frac{1}{2}ln(|\sigma^2 I|) - \frac{1}{2}(x-\alpha s)^T(\sigma^2 I)^{-1}(x-\alpha s)\,, \tag{2.9}$$

$$\frac{\partial ln(p(x|H_1))}{\partial \alpha} = s^T(\sigma^2 I)^{-1}x - \hat{\alpha} s^T(\sigma^2 I)^{-1}s = 0\,, \tag{2.10}$$

$$\hat{\alpha} = \frac{s^T x}{s^T s}\,. \tag{2.11}$$

Applying GLRT to log-likelihood functions and inserting the estimated parameters which are given in Equation 2.8 and 2.11, the simple form of the SAM detector can be obtained:

$$\begin{aligned}\Lambda(x) = \frac{ln(p(x|H_1))}{ln(p(x|H_0))} &= ln(p(x|H_1)) - ln(p(x|H_0)) \\ &= \frac{x^T x}{\hat{\sigma}^2} - \frac{(x-\hat{\alpha}s)^T(x-\hat{\alpha}s)}{\hat{\sigma}^2} \\ &= \frac{(s^T x)^2}{\hat{\sigma}^2(s^T s)} = \frac{(s^T x)^2 L}{(x^T x)(s^T s)}\end{aligned} \tag{2.12}$$

The SAM detector can be formulated by ignoring the scalar value *L*, taking the square-root of the result given in Equation 2.12 and applying the inverse cosine:

$$T_{\text{SAM}}(x) = arccos\left(\frac{s^T x}{(s^T s)^{\frac{1}{2}}(x^T x)^{\frac{1}{2}}}\right). \tag{2.13}$$

The assumption for the zero-mean white Gaussian background in SAM method cannot be appropriate for every scene. Therefore, satisfactory results cannot be obtained when SAM is applied to counter examples of this assumption. SAM can be thought as a primitive operation for target detection due to applying only vector comparison. Also, it can be used as a spectral distance metric for the superpixel extraction shown in Chapter 4.

2.2.2 Spectral Matched Filter (SMF)

SAM is not enough to solve target detection problem due to the assumption of zero-mean and white noise background. Non-zero mean background assumption is base of the Spectral Matched Filter (SMF) algorithm. The hypotheses of SMF are shown as

$$\begin{aligned} H_0 &: x = b \sim N(\mu, \Sigma), \\ H_1 &: x = \alpha s + b \sim N(\alpha s + \mu, \Sigma) \end{aligned} \tag{2.14}$$

where μ is non-zero mean value of the background. The unknown parameters are α, μ and $\mathit{\Sigma}$. The probability distribution functions of both hypotheses for SMF can be expressed as

$$p(x|H_0) = \frac{1}{(2\pi)^{\frac{L}{2}}|\hat{\Sigma}|^{\frac{1}{2}}} exp\left\{ -\frac{1}{2}(x - \hat{\mu})^T(\hat{\Sigma})^{-1}(x - \hat{\mu}) \right\}, \tag{2.15}$$

$$p(x|H_1) = \frac{1}{(2\pi)^{\frac{L}{2}}|\hat{\Sigma}|^{\frac{1}{2}}} exp\left\{ -\frac{1}{2}(x - \hat{\alpha}s - \hat{\mu})^T(\hat{\Sigma})^{-1}(x - \hat{\alpha}s - \hat{\mu}) \right\}. \tag{2.16}$$

The likelihood ratio can be found by using the hypotheses given in Equation 2.14:

$$\begin{aligned} \Lambda(x) = \frac{ln(p(x|H_1))}{ln(p(x|H_0))} &= ln(p(x|H_1)) - ln(p(x|H_0)) = \\ \hat{\alpha}s^T\hat{\Sigma}^{-1}x - &\frac{1}{2}\hat{\alpha}^2 s^T\hat{\Sigma}^{-1}s - \hat{\alpha}s^T\hat{\Sigma}^{-1}\hat{\mu}. \end{aligned} \tag{2.17}$$

The MLE for α is obtained from the Equation 2.16 by simply taking derivative of the log-likelihood function and equating it to zero. Estimated α value is calculated as

$$\hat{\alpha} = \frac{s^T\hat{\Sigma}^{-1}(x - \hat{\mu})}{s^T\hat{\Sigma}^{-1}s}, \tag{2.18}$$

and this parameter is inserted into Equation 2.17 to obtain the likelihood ratio:

$$\Lambda(x) = \frac{1}{2}\frac{[s^T\hat{\Sigma}^{-1}(x - \hat{\mu})]^2}{s^T\hat{\Sigma}^{-1}s}. \tag{2.19}$$

Ignoring scalar value which does not affect the result of the detector, the SMF detector can be expressed as

$$T_{\text{SMF}}(x) = \frac{[s^T\hat{\Sigma}^{-1}(x - \hat{\mu})]^2}{s^T\hat{\Sigma}^{-1}s}. \tag{2.20}$$

$\mathit{\Sigma}$ can be estimated by taking derivative with respect to $\mathit{\Sigma}$ of the log-likelihood function and equating the result to zero. The resulting formulation of SMF which is shown in Equation 2.20 has some problems [23]. For the case of α=*1* in the target-present hypothesis given in Equation 2.14, the expected value of *x* should be equal to *s* instead of s+μ. For this reason, background mean value can be subtracted from target spectra in the SMF formulation. Also there is no constraint to the parameter α in the SMF detector. Although it is not possible to obtain negative α value physically, taking the positive square-root of the resulting formulation is more reasonable for the mathematical expression. After these two important modifications the SMF detector can be obtained as

$$T_{\text{SMF}}(x) = \frac{(s-\hat{\mu})^T \hat{\Sigma}^{-1}(x-\hat{\mu})}{\sqrt{(s-\hat{\mu})^T \hat{\Sigma}^{-1}(s-\hat{\mu})}}. \tag{2.21}$$

The above mentioned derivation of SMF detector are based on three main assumptions [22]: i) the scaled target spectrum α*s* is added to background spectrum *b*, ii) the background has the same mean value μ under both target-absent and target-present hypotheses, and iii) the covariance matrix of target and background is the same. Although all of these assumptions are invalid for real hyperspectral images, development of detection algorithms is provided based on this model.

2.2.3 Constrained Energy Minimization (CEM)

Constrained Energy Minimization (CEM) is proposed as a mean-centered version of SMF. CEM uses finite impulse response filter and minimizes the filter output energy while constraining the desired target signature by a specific gain [25]. It can be obtained from general linear filter function

$$r(x) = h^T x, \tag{2.22}$$

where *h* is the impulse response of the filter and *x* is the image pixel. This function should be optimized by minimizing the background energy function. The background energy function can be formulated as

$$\begin{aligned} E &= \frac{1}{N}\sum_{i=1}^{N} r^2(x_\mathrm{i}) \\ &= \frac{1}{N}\sum_{i=1}^{N} h^T x_\mathrm{i} {x_\mathrm{i}}^T h \\ &= h^T \left(\frac{1}{N}\sum_{i=1}^{N} x_\mathrm{i} {x_\mathrm{i}}^T \right) h \\ &= h^T R h \end{aligned} \tag{2.23}$$

where *R* is the sample correlation matrix and *N* is the number of pixels in the hyperspectral image. The optimization problem can be defined as

$$\min_h \; h^T R h \;\; \text{subject to} \;\; h^T s = 1 \tag{2.24}$$

where *s* is the target spectrum. To simplify the problem a factor of 1/2 is multiplied and constraint function with Lagrange multiplier is added to the energy formula [26]:

$$\min_h \; \left\{ \frac{1}{2} h^T R h + \lambda (h^T s - 1) \right\}. \tag{2.25}$$

To obtain minimized energy function, the derivative of the Equation 2.25 with respect to *h* is taken and equated to zero:

$$\begin{aligned} \frac{dE}{dh} &= Rh + \lambda s = 0 \\ -Rh &= \lambda s \\ h &= -\lambda R^{-1} s \end{aligned} \tag{2.26}$$

Lagrange multiplier λ value can be found by inserting h vector into the Equation 2.26 and using the constraint function:

$$\begin{aligned} s^T h &= 1 \\ -s^T R^{-1} s \lambda &= 1 \\ \lambda &= -[s^T R^{-1} s]^{-1} \end{aligned} \tag{2.27}$$

After finding Lagrange multiplier value with respect to the target signature and the sample correlation matrix, final CEM formulation can be found by inserting λ value into Equation 2.26:

$$h = \frac{R^{-1} s}{s^T R^{-1} s} \tag{2.28}$$

$$T_{\text{CEM}}(x) = \frac{s^T R^{-1} x}{s^T R^{-1} s} \tag{2.29}$$

2.2.4 Adaptive Coherence Estimator (ACE)

Adaptive Coherence Estimator (ACE) algorithm models the background by multivariate normal distribution and uses the GLRT approach for the detection. Background and target pixels have different scaled covariance matrices. The hypotheses of ACE are shown as

$$\begin{aligned} H_0 &: x = b \sim N(0, {\sigma_0}^2\Sigma), \\ H_1 &: x = \alpha s + \beta b \sim N(\alpha s, {\sigma_1}^2\Sigma) \end{aligned} \tag{2.30}$$

where ${\sigma_0}^2$ and ${\sigma_1}^2$ are unknown scaling terms and α, β are unknown parameters. Unknown parameters are estimated as the same way in the other detection algorithms by using MLE. Probability distribution functions of both hypotheses can be formulated as

$$p(x|H_0) = \frac{1}{(2\pi\hat{\sigma_0}^2)^{\frac{L}{2}}|\Sigma|^{\frac{1}{2}}} exp\Bigg\{ -\frac{1}{2\hat{\sigma_0}^2} x^T\Sigma^{-1}x \Bigg\}, \tag{2.31}$$

$$p(x|H_1) = \frac{1}{(2\pi\hat{\sigma_1}^2)^{\frac{L}{2}}|\Sigma|^{\frac{1}{2}}} exp\Bigg\{ -\frac{1}{2\hat{\sigma_1}^2} (x - \hat{\alpha}s)^T\Sigma^{-1}(x - \hat{\alpha}s) \Bigg\}. \tag{2.32}$$

Log likelihood functions can be expressed as

$$ln(p(x|H_0)) = -\frac{L}{2}ln(2\pi{\sigma_0}^2) - \frac{1}{2}ln(|\Sigma|) - \frac{1}{2{\sigma_0}^2}x^T\Sigma^{-1}x\,, \tag{2.33}$$

$$ln(p(x|H_1)) = -\frac{L}{2}ln(2\pi{\sigma_1}^2) - \frac{1}{2}ln(|\Sigma|) - \frac{1}{2{\sigma_1}^2}(x - \alpha s)^T\Sigma^{-1}(x - \alpha s)\,. \tag{2.34}$$

Estimated parameters can be found as

$$\frac{\partial ln(p(x|H_1))}{\partial\alpha} = \frac{1}{{\sigma_1}^2}[s^T\Sigma^{-1}x - \hat{\alpha}s^T\Sigma^{-1}s] = 0 \Rightarrow \hat{\alpha} = (s^T\Sigma^{-1}s)^{-1}s^T\Sigma^{-1}x \tag{2.35}$$

$$\frac{\partial ln(p(x|H_0))}{\partial{\sigma_0}^2} = -\frac{L}{2\hat{\sigma_0}^2} + \frac{1}{2\hat{\sigma_0}^4}x^T\Sigma^{-1}x = 0 \Rightarrow \hat{\sigma_0}^2 = \frac{1}{L}x^T\Sigma^{-1}x \tag{2.36}$$

$$\begin{gathered} \frac{\partial ln(p(x|H_1))}{\partial{\sigma_1}^2} = -\frac{L}{2\hat{\sigma_1}^2} + \frac{1}{2\hat{\sigma_1}^4}(x - \alpha s)^T\Sigma^{-1}(x - \alpha s) = 0 \Rightarrow \\ \hat{\sigma_1}^2 = \frac{1}{L}(x - \alpha s)^T\Sigma^{-1}(x - \alpha s). \end{gathered} \tag{2.37}$$

To obtain the ACE detector, these estimated parameters are substituted back into GLRT equation.

$$GLRT(x) = \left(\frac{\hat{\sigma_1}^2}{\hat{\sigma_0}^2}\right)^{-\frac{L}{2}} exp\Bigg\{ -\frac{1}{2\hat{\sigma_1}^2}(x - \hat{\alpha}s)^T\Sigma^{-1}(x - \hat{\alpha}s) + \frac{1}{2\hat{\sigma_0}^2}x^T\Sigma^{-1}x \Bigg\} \tag{2.38}$$

$$GLRT(x) = \left(1 - \frac{x^T\Sigma^{-1}s(s^T\Sigma^{-1}s)^{-1}s^T\Sigma^{-1}x}{x^T\Sigma^{-1}x}\right)^{-\frac{L}{2}} \tag{2.39}$$

The right-sided term of the Equation 2.39 is monotonically increasing function which is statistically equivalent to the equation given below [27]. The ACE algorithm can be formulated as

$$T_{\text{ACE}}(x) = \frac{x^T\Sigma^{-1}s(s^T\Sigma^{-1}s)^{-1}s^T\Sigma^{-1}x}{x^T\Sigma^{-1}x}. \tag{2.40}$$

The ACE algorithm has the Constant False Alarm Rate (CFAR) property and is invariant to scaling of data. Therefore, ACE provides more realistic model which is called as unstructured background modeling and reveals better performances compared to its counterparts [21].

2.2.5 Adaptive Matched Subspace Detector (AMSD)

Adaptive Matched Subspace Detector (AMSD) algorithm uses GLRT for detection and requires the background signature with target signature as inputs. This algorithm models the background by using linear mixture model [12]. The hypotheses of AMSD are given below

$$\begin{aligned} H_0 &: x = \alpha_{\text{b},0}b + n \sim N(\alpha_{\text{b},0}b, {\sigma_0}^2 I), \\ H_1 &: x = \alpha e + n = \alpha_{\text{s}}s + \alpha_{\text{b},1}b + n \sim N(\alpha_{\text{s}}s + \alpha_{\text{b},1}b, {\sigma_1}^2 I) \end{aligned} \tag{2.41}$$

where *e* is a matrix that contains the concatenation of target and background signatures. The probability distribution functions of both hypotheses can be given as

$$p(x|H_0) = \frac{1}{(2\pi\hat{\sigma_0}^2)^{\frac{L}{2}}}\, exp\left\{ -\frac{1}{2\hat{\sigma_0}^2}(x - \hat{\alpha}_{\text{b},0}b)^T(x - \hat{\alpha}_{\text{b},0}b)\right\}, \tag{2.42}$$

$$p(x|H_1) = \frac{1}{(2\pi\hat{\sigma_1}^2)^{\frac{L}{2}}}\, exp\left\{ -\frac{1}{2\hat{\sigma_1}^2}(x - \hat{\alpha}e)^T(x - \hat{\alpha}e)\right\}. \tag{2.43}$$

Log-likelihood functions of both hypotheses are

$$ln(p(x|H_0)) = -\frac{L}{2}ln(2\pi) - \frac{L}{2}ln({\sigma_0}^2) - \frac{1}{2{\sigma_0}^2}(x - \alpha_{\text{b},0}b)^T(x - \alpha_{\text{b},0}b), \tag{2.44}$$

$$ln(p(x|H_1)) = -\frac{L}{2}ln(2\pi) - \frac{L}{2}ln({\sigma_1}^2) - \frac{1}{2{\sigma_1}^2}(x - \alpha e)^T(x - \alpha e). \tag{2.45}$$

The unknown parameters are estimated by using MLE as shown below:

$$\frac{\partial ln(p(x|H_0))}{\partial \alpha_{\text{b},0}} = \frac{1}{{\sigma_0}^2}[b^Tx - \hat{\alpha}_{\text{b},0}b^Tb] = 0 \Rightarrow \hat{\alpha}_{\text{b},0} = (b^Tb)^{-1}b^Tx \tag{2.46}$$

$$\begin{aligned}\frac{\partial ln(p(x|H_0))}{\partial {\sigma_0}^2} &= -\frac{L}{2\hat{\sigma_0}^2} + \frac{1}{2\hat{\sigma_0}^4}(x - \alpha_{\mathrm{b},0}b)^T(x - \alpha_{\mathrm{b},0}b) = 0 \Rightarrow \\ \hat{\sigma_0}^2 &= \frac{1}{L}(x - \alpha_{\mathrm{b},0}b)^T(x - \alpha_{\mathrm{b},0}b).\end{aligned} \tag{2.47}$$

Similarly by applying MLE, the unknown parameters of target-present hypothesis can be found like that:

$$\hat{\alpha} = (e^T e)^{-1} e^T x \tag{2.48}$$

$$\hat{\sigma_1}^2 = \frac{1}{L}(x - \alpha e)^T(x - \alpha e). \tag{2.49}$$

Equations shown below are obtained by rewriting the estimated parameters which are shown in Equations 2.46, 2.47, 2.48, and 2.49 into the probability distribution functions of hypotheses:

$$\begin{aligned}p(x|H_0) &= \Bigg(\frac{2\pi}{L}(x - b(b^T b)^{-1} b^T x)^T (x - b(b^T b)^{-1} b^T x)\Bigg)^{-\frac{L}{2}} exp\Bigg\{-\frac{L}{2}\Bigg\} \\ &= \Bigg(\frac{2\pi}{L} x^T (I - b(b^T b)^{-1} b^T) x\Bigg)^{-\frac{L}{2}} exp\Bigg\{-\frac{L}{2}\Bigg\},\end{aligned} \tag{2.50}$$

$$\begin{aligned}p(x|H_1) &= \Bigg(\frac{2\pi}{L}(x - e(e^T e)^{-1} e^T x)^T (x - e(e^T e)^{-1} e^T x)\Bigg)^{-\frac{L}{2}} exp\Bigg\{-\frac{L}{2}\Bigg\} \\ &= \Bigg(\frac{2\pi}{L} x^T (I - e(e^T e)^{-1} e^T) x\Bigg)^{-\frac{L}{2}} exp\Bigg\{-\frac{L}{2}\Bigg\}.\end{aligned} \tag{2.51}$$

The GLRT can be found by using the likelihood functions shown in Equations 2.50 and 2.51 and applying simple algebra:

$$\frac{p(x|H_0)}{p(x|H_1)} = \frac{x^T (I - b(b^T b)^{-1} b^T) x}{x^T (I - e(e^T e)^{-1} e^T) x} = \frac{x^T {P_{\mathrm{B}}}^{\perp} x}{x^T {P_{\mathrm{Z}}}^{\perp} x}. \tag{2.52}$$

The Equation 2.52 is modified to obtain AMSD due to difficulty of identifying the distribution of the detection statistic. The AMSD algorithm can be expressed as

$$T_{\mathrm{AMSD}}(x) = \frac{x^T ({P_{\mathrm{B}}}^{\perp} - {P_{\mathrm{Z}}}^{\perp}) x}{x^T {P_{\mathrm{Z}}}^{\perp} x}. \tag{2.53}$$

AMSD algorithm has the CFAR property like the ACE algorithm. While AMSD algorithm is based on the structured background that employs the endmembers to obtain the background, ACE detector is based on the unstructured background that models the background by a statistical distribution instead of endmembers.

2.3 Subspace Projection-Based Methods

The known background spectra are used to model the background by a subspace model in OSP and TCIMF algorithms and the unknown background spectra are obtained by applying orthogonal subspace projection to target spectrum in DTDCA algorithm. All of these algorithms can be described as the geometrical approaches to the target detection problem. The theories and the procedures of these algorithms are given in this section.

2.3.1 Target-Constrained Interference-Minimized Filter (TCIMF)

Target-Constrained Interference-Minimized Filter (TCIMF) is extended version of the CEM detector by usage of non-target spectra besides the target spectrum [28]. To detect desired targets, to minimize the interfering effects of background and to eliminate the non-targets are the aims of the TCIMF. The theorem behind it is very similar to CEM detector. TCIMF detector is obtained by using general linear filter function which can be shown as

$$r(x) = h^T x. \tag{2.54}$$

Let $s = [s_1, s_2, ..., s_{\mathrm{L}}]^T$ and $B = \begin{bmatrix} b_{11} & b_{12} & b_{13} & \dots & b_{1L} \\ b_{21} & b_{22} & b_{23} & \dots & b_{2L} \\ \dots & \dots & \dots & \dots & \dots \\ b_{q1} & b_{q2} & b_{q3} & \dots & b_{qL} \end{bmatrix}^T$ denote the desired target signature vector and non-target signature matrix, respectively. In accordance with these inputs, a constraint vector can be defined as

$$[s_{\mathrm{Lx1}}\, B_{\mathrm{Lxq}}]^T h = \begin{bmatrix} 1 \\ 0_{\mathrm{qx1}} \end{bmatrix} \tag{2.55}$$

where q represents the number of non-target signatures and L is the number of spectral bands. The optimization problem is shown as

$$\min_h \; h^T R h \;\text{ subject to }\; [s\, B]^T h = \begin{bmatrix} 1 \\ 0_{\mathrm{qx1}} \end{bmatrix}. \tag{2.56}$$

Energy function can be expressed as

$$\begin{aligned} E &= \frac{1}{L}\sum_{i=1}^{L} r^2(x_\mathrm{i}) \\ &= \frac{1}{L}\sum_{i=1}^{L} h^T x_\mathrm{i} {x_\mathrm{i}}^T h \\ &= h^T\left(\frac{1}{L}\sum_{i=1}^{L} x_\mathrm{i} {x_\mathrm{i}}^T\right)h \\ &= h^T R h. \end{aligned} \tag{2.57}$$

The optimization problem is defined as

$$\min_h \left\{\frac{1}{2}h^T R h + \lambda\left([s\,B]^T h - \begin{bmatrix} 1 \\ 0_{\mathrm{qx1}} \end{bmatrix}\right)\right\}. \tag{2.58}$$

To obtain minimized energy function, the derivative of the Equation 2.58 with respect to *h* is taken and equated to zero

$$\begin{aligned} \frac{dE}{dh} &= h^T R + \lambda [s\,B]^T = [0_{\mathrm{Lx1}}] \\ -h^T R &= \lambda [s\,B]^T \\ h^T &= -\lambda [s\,B]^T R^{-1} \Rightarrow \\ h &= -R^{-1}[s\,B]\lambda^T, \end{aligned} \tag{2.59}$$

where λ^T is a vector whose size (q+1)x1. Lagrange multiplier vector λ^T can be found by inserting h vector into the Equation 2.59 and using constraint function:

$$\begin{aligned} -[s\,B]^T R^{-1}[s\,B]\lambda^T &= \begin{bmatrix} 1 \\ 0_{\mathrm{qx1}} \end{bmatrix} \\ \lambda^T &= -\Big([s\,B]^T R^{-1}[s\,B]\Big)^{-1}\begin{bmatrix} 1 \\ 0_{\mathrm{qx1}} \end{bmatrix} \end{aligned} \tag{2.60}$$

TCIMF formulation can be obtained by inserting λ^T vector shown in Equation 2.60 into Equation 2.59:

$$h = R^{-1}[s\,B]\Big([s\,B]^T R^{-1}[s\,B]\Big)^{-1}\begin{bmatrix} 1 \\ 0_{\mathrm{qx1}} \end{bmatrix} \tag{2.61}$$

$$T_{\mathrm{TCIMF}}(x) = R^{-1}[s\,B]\Big([s\,B]^T R^{-1}[s\,B]\Big)^{-1}\begin{bmatrix} 1 \\ 0_{\mathrm{qx1}} \end{bmatrix} x. \tag{2.62}$$

2.3.2 Orthogonal Subspace Projection (OSP)

Orthogonal Subspace Projection (OSP) is a detector which is used for finding the target in the subspace orthogonal to the background subspace by maximizing the signal-to-noise ratio (SNR) [8]. Structured subspace model is used to characterize spectral variability in the data. The spectral signature of the background should be known for OSP. The target-present and target-absent hypotheses of the OSP are given below.

$$\begin{aligned} H_0 &: x = \beta b + n \sim N(\beta b, {\sigma_0}^2 I), \\ H_1 &: x = \alpha s + \beta b + n \sim N(\alpha s + \beta b, {\sigma_1}^2 I). \end{aligned} \tag{2.63}$$

Probability distribution functions of these hypotheses can be given as

$$p(x|H_0) = \frac{1}{(2\pi\hat{\sigma_0}^2)^{\frac{L}{2}}} exp\Bigg\{ -\frac{1}{2\hat{\sigma_0}^2}(x-\hat{\beta}b)^T(x-\hat{\beta}b)\Bigg\}, \tag{2.64}$$

$$p(x|H_1) = \frac{1}{(2\pi\hat{\sigma_1}^2)^{\frac{L}{2}}} exp\Bigg\{ -\frac{1}{2\hat{\sigma_1}^2}(x-\hat{\alpha}s-\hat{\beta}b)^T(x-\hat{\alpha}s-\hat{\beta}b)\Bigg\}. \tag{2.65}$$

Log-likelihood functions of both hypotheses are

$$ln(p(x|H_0)) = -\frac{L}{2}ln(2\pi) - \frac{L}{2}ln({\sigma_0}^2) - \frac{1}{2{\sigma_0}^2}(x-\hat{\beta}b)^T(x-\hat{\beta}b), \tag{2.66}$$

$$ln(p(x|H_1)) = -\frac{L}{2}ln(2\pi) - \frac{L}{2}ln({\sigma_1}^2) - \frac{1}{2{\sigma_1}^2}(x-\hat{\alpha}s-\hat{\beta}b)^T(x-\hat{\alpha}s-\hat{\beta}b). \tag{2.67}$$

The unknown parameters of both hypotheses can be found by using MLE. Also, $\hat{\beta}$ and $\hat{\alpha}$ can be found by using the least-square algorithm which are shown below.

$$\begin{aligned} \frac{\partial ln(p(x|H_0))}{\partial\hat{\beta}} &= \frac{1}{{\sigma_0}^2}[b^Tx - \hat{\beta}b^Tb] = 0 \Rightarrow \hat{\beta} = (b^Tb)^{-1}b^Tx, \\ \min_{\beta}(x-\beta b)^2 \Rightarrow \frac{\partial(x-\hat{\beta}b)^2}{\partial\hat{\beta}} &= -2b^Tx + 2\hat{\beta}b^Tb = 0 \\ &\Rightarrow \hat{\beta}_{\text{LS}} = (b^Tb)^{-1}b^Tx. \end{aligned} \tag{2.68}$$

$$\begin{aligned} \frac{\partial ln(p(x|H_1))}{\partial\hat{\alpha}} &= \frac{1}{{\sigma_1}^2}[s^Tx - \hat{\alpha}s^Ts - s^Tb\hat{\beta}] = 0 \Rightarrow \hat{\alpha} = \frac{s^Tx - s^Tb\hat{\beta}}{s^Ts} \\ \Rightarrow \hat{\alpha} &= \frac{s^Tx - s^Tb(b^Tb)^{-1}b^Tx}{s^Ts} \Rightarrow \hat{\alpha} = \frac{s^T(I-b(b^Tb)^{-1}b^T)x}{s^Ts}, \\ \min_{\alpha}(x-\alpha s-\beta b)^2 \Rightarrow \frac{\partial(x-\hat{\alpha}s-\hat{\beta}b)^2}{\partial\hat{\alpha}} &= -2s^Tx + 2\hat{\alpha}s^Ts - 2\hat{\beta}s^Tb = 0 \\ \Rightarrow \hat{\alpha} &= \frac{s^Tx - s^Tb\hat{\beta}}{s^Ts} = \frac{s^Tx - s^Tb(b^Tb)^{-1}b^Tx}{s^Ts} \\ \Rightarrow \hat{\alpha}_{\text{LS}} &= \frac{s^T(I-b(b^Tb)^{-1}b^T)x}{s^Ts}. \end{aligned} \tag{2.69}$$

Subspace projection operator P_b can be shown as

$$P_\text{b} = b(b^T b)^{-1} b^T \tag{2.70}$$

and its orthogonal complement is formulated as

$$P_\text{b}^{\perp} = I - b(b^T b)^{-1} b^T. \tag{2.71}$$

Subspace projection operator has some properties such as

$$\begin{aligned} P_\text{b} &= P_\text{b}^T, \\ P_\text{b} &= P_\text{b}^T P_\text{b}, \\ P_\text{b}^{\perp} &= (P_\text{b}^{\perp})^T P_\text{b}^{\perp}. \end{aligned} \tag{2.72}$$

By using these properties $\hat{\alpha}$ can be formulated as

$$\hat{\alpha} = \frac{s^T P_\text{b}^{\perp} x}{s^T P_\text{b}^{\perp} s} \tag{2.73}$$

The estimated abundance of the target can be used for detection statistic. Therefore, OSP detector can be expressed as

$$T_\text{OSP}(x) = \frac{s^T P_\text{b}^{\perp} x}{s^T P_\text{b}^{\perp} s}. \tag{2.74}$$

OSP is based on linear mixture model like other detection algorithms. After separating background from the desired target in the mixture model, orthogonal subspace projector is applied to the undesired background signatures to eliminate them and detect the desired target.

2.3.3 Desired Target Detection and Classification Algorithm (DTDCA)

Desired Target Detection and Classification Algorithm (DTDCA) is a member of the subspace projection-based target detection algorithms. Most of the time it is difficult to obtain prior knowledge of the undesired signatures. Therefore, DTDCA algorithm can be used for this type of applications. DTDCA procedure can be summarized as follows [29].

1. Desired target signature is selected as an initial signature and shown as s_0. Error threshold value ε is defined. $i = 0$ and $b_0 = \emptyset$.

2. Apply $P_{\mathrm{s_0}}{}^{\perp}$ to all image pixel vectors **x** to obtain orthogonal subspace projection ($P_{\mathrm{s_0}}{}^{\perp} = I - s_0(s_0{}^T s_0)^{-1} s_0{}^T$).

3. The first undesired signature is obtained by equating it to maximum orthogonal projection

   $$s_1 = arg\left\{ \max_{x}[(P_{\mathrm{s_0}}{}^{\perp}x)^T(P_{\mathrm{s_0}}{}^{\perp}x)] \right\}. \tag{2.75}$$

   Set $i = 1$ and $b_1 = s_1$.

4. If $\eta_1 = s_0{}^T P_{\mathrm{b_1}}{}^{\perp} s_0 < \varepsilon$, go to step 8. Otherwise, increase i by 1 and continue.

5. The other undesired signatures are obtained by

   $$s_{\mathrm{i}} = arg\left\{ \max_{x}[(P_{[\mathrm{s_0 b_{i\text{-}1}}]}{}^{\perp}x)^T(P_{[\mathrm{s_0 b_{i\text{-}1}}]}{}^{\perp}x)] \right\} \tag{2.76}$$

   where $b_{\mathrm{i\text{-}1}} = [s_1\, s_2 \ldots s_{\mathrm{i\text{-}1}}]$ is the undesired signature matrix.

6. $b_{\mathrm{i}} = [s_1\, s_2 \ldots s_{\mathrm{i}}]$ is the *i*th undesired signature matrix. Orthogonal Projection Correlation Index (OPCI) η_{i} is calculated by the given formula below and the result is compared to error threshold value ε.

   $$\eta_{\mathrm{i}} = s_0{}^T P_{\mathrm{b_i}}{}^{\perp} s_0 \tag{2.77}$$

7. If $\eta_{\mathrm{i}} > \varepsilon$, go to step 5 to calculate new signature. Otherwise, continue.

8. As a result, undesired signature matrix b_{i} is obtained at this step.

9. At the target classification process, use b_{i} matrix for OSP detector to classify all image pixels.

   $$T_{\mathrm{DTDCA}}(x) = s_0{}^T P_{\mathrm{b_i}}{}^{\perp} x \tag{2.78}$$

As a summary of all signature-based target detection methods, Table 4.2, which is an enhanced version of the table given in [23], is shown below. BG is the abbreviation of the background.

Algorithm	**Models**			**Parameters**	
	Signal	Background	Noise	Known	Unknown
SAM	Deterministic	White	Part of BG	s	α,σ
SMF	Deterministic	Normal	Part of BG	s	α,μ,Σ
CEM	Deterministic	Normal	Part of BG	s	R
ACE	Deterministic	Normal	Part of BG	s	α,β,Σ
AMSD	Deterministic	-	White	s,b	$\alpha_{\mathrm{b},0}$,α,σ_0,σ_1
TCIMF	Deterministic	-	Part of BG	s,b	R
OSP	Deterministic	Subspace	White	s,b	α,β,σ_0,σ_1
DTDCA	Deterministic	Subspace	White	s_0	b

Table 2.1: Taxonomy of Signature-Based Target Detection Algorithms

2.4 Hybrid Target Detection Methods

Hybrid detectors are based on a different idea that can be described as modeling the background by linear mixing model with a statistical hypothesis test. Modeling the background physically and statistically, improves the performance compared to other approaches due to consideration of both effects [12]. Linear unmixing algorithm and the target detection algorithms are two main parts of the hybrid detectors. As a linear unmixing algorithm Fully-Constrained Least Squares (FCLS) algorithm and two different hybrid detector algorithms, which are Hybrid Structured Detector (HSD) and Hybrid Unstructured Detector (HUD), are described in this section. The FCLS algorithm is utilized to estimate the abundances of endmembers by meeting the abundance constraints due to the high computational efficiency compared to other linear unmixing algorithms [12].

2.4.1 Fully-Constrained Least Squares (FCLS) Algorithm

Linear unmixing is one of the fundamental methods which are used for hyperspectral target detection algorithms. The main purpose of the linear unmixing is accurate estimation of the abundances of endmembers. An image pixel contains endmembers with different abundances as an assumption of the linear mixing model. Linear mixing model for *p* endmembers and *l* spectral bands can be shown as

$$x = M\alpha + n \tag{2.79}$$

where *x* is image pixel vector, *M* is an *lxp* signature matrix of *p* endmembers, α is a *px1* abundance vector and *n* is a *lx1* additive white Gaussian noise vector. There can be suboptimal solutions when the linear unmixing model is unconstrained. To overcome this problem, FCLS was proposed [30]. Two constraints of the FCLS can be shown as

$$\sum_{j=1}^{p} \alpha_{\mathrm{j}} = 1 \ and \ \alpha_{\mathrm{j}} \geq 0 \ for \ all \ 1 \leq j \leq p \tag{2.80}$$

which are called sum-to-one constraint and non-negativity constraint respectively. FCLS procedure for hyperspectral images can be summarized as follows:

1. Constrained least-squares linear mixing is an optimization problem which energy function can be written as

   $$J = \frac{1}{2}(x - M\alpha)^T (x - M\alpha) - \lambda\left(\sum_{j=1}^{p} \alpha_{\mathrm{j}} - 1\right) \tag{2.81}$$

   MLE of the α parameter can be found by differentiating Equation 2.81 with respect to α and equating the result to zero:

   $$\begin{aligned} \frac{\partial J}{\partial \alpha} &= -M^T x + M^T M \hat{\alpha}_{\mathrm{FCLS}} - \lambda = 0 \Rightarrow \\ \hat{\alpha}_{\mathrm{FCLS}} &= (M^T M)^{-1}(M^T x + \lambda). \end{aligned} \tag{2.82}$$

   Least-square solution of the Equation 2.79 can be found as

   $$\min_{\alpha}(x - M\alpha)^2 \Rightarrow \hat{\alpha_{\mathrm{LS}}} = (M^T M)^{-1} M^T x. \tag{2.83}$$

2. Inserting $\hat{\alpha}_{\mathrm{LS}}$ into Equation 2.82 and applying the sum-to-one constraint to $\hat{\alpha}_{\mathrm{FCLS}}$, $\hat{\alpha}_{\mathrm{FCLS}}$ and λ parameters can be found as

   $$\begin{aligned} \hat{\alpha}_{\mathrm{FCLS}} &= \hat{\alpha}_{\mathrm{LS}} + \lambda (M^T M)^{-1} \\ \lambda &= \frac{\hat{\alpha}_{\mathrm{FCLS}} - \hat{\alpha}_{\mathrm{LS}}}{(M^T M)^{-1}} = \frac{1 - \mathbf{1}^T \hat{\alpha}_{\mathrm{LS}}}{\mathbf{1}^T (M^T M)^{-1}} \end{aligned} \tag{2.84}$$

3. Check the elements of $\hat{\alpha}_{\mathrm{FCLS}}$. If all of them are positive, the algorithm stops. Otherwise, go to step 4.

4. Find the maximum absolute value of $|\hat{\alpha}_{\text{FCLS,j}}/(M^TM)_{\text{j}}{}^{-1}|$ term where $\hat{\alpha}_{\text{FCLS,j}}$ corresponds to negative elements of $\hat{\alpha}_{\text{FCLS}}$ vector and the corresponding component in $(M^TM)^{-1}$ vector shown as $(M^TM)_{\text{j}}{}^{-1}$. Then remove the endmember signature in M matrix which corresponds to this component. Go to step 1 and re-apply this procedure with new M matrix.

As a result, abundances of the endmembers are obtained by considering the constraints. These constraints are not always applicable to real world, therefore another constrained linear unmixing method such as Non-negativity Constrained Least Squares (NCLS) algorithm and Fast Non-negativity Constrained Least Squares (FNNLS) algorithm [31, 32] can be used due to applying only non-negativity constraint.

2.4.2 Hybrid Structured Detector (HSD)

The HSD algorithm uses both structured background, which is similar to AMSD algorithm, and linear unmixing algorithm. Applying these algorithms together provides physically and statistically meaningful analysis of the scene [12]. Hypotheses of HSD are

$$\begin{aligned} H_0 &: x = \alpha_{\text{b,0}} b + n \sim N(\alpha_{\text{b,0}} b, {\sigma_0}^2 \Sigma) \\ H_1 &: x = \alpha e + n = \alpha_{\text{s}} s + \alpha_{\text{b,1}} b + n \sim N(\alpha_{\text{s}} s + \alpha_{\text{b,1}} b, {\sigma_1}^2 \Sigma). \end{aligned} \tag{2.85}$$

The likelihood equations can be shown as

$$p(x|H_0) = \frac{1}{(2\pi\hat{\sigma_0}^2)^{\frac{L}{2}}|\Sigma|^{\frac{1}{2}}} exp\left\{ -\frac{1}{2\hat{\sigma_0}^2}(x - \hat{\alpha}_{\text{b,0}} b)^T \Sigma^{-1} (x - \hat{\alpha}_{\text{b,0}} b) \right\}, \tag{2.86}$$

$$p(x|H_1) = \frac{1}{(2\pi\hat{\sigma_1}^2)^{\frac{L}{2}}|\Sigma|^{\frac{1}{2}}} exp\left\{ -\frac{1}{2\hat{\sigma_1}^2}(x - \hat{\alpha} e)^T \Sigma^{-1} (x - \hat{\alpha} e) \right\} \tag{2.87}$$

where $\alpha e = \alpha_{\text{s}} s + \alpha_{\text{b,1}} b$.

MLE of the unknown parameters can be found as

$$\hat{\sigma_0}^2 = \frac{1}{L}(x - \alpha_{\text{b,0}} b)^T \Sigma^{-1} (x - \alpha_{\text{b,0}} b), \tag{2.88}$$

$$\hat{\sigma_1}^2 = \frac{1}{L}(x - \alpha e)^T \Sigma^{-1} (x - \alpha e). \tag{2.89}$$

Additionally, the abundance estimation should be calculated by using FCLS algorithm instead of MLE. The optimization problem can be defined as

$$\min_{\alpha}(x-\alpha e)^T\Sigma^{-1}(x-\alpha e)\ with\,constraints\ \sum_{i=1}^{p}\alpha_{\mathrm{i}}=1\,and\,\alpha_{\mathrm{i}}\geq 0. \tag{2.90}$$

The energy function can be formulated as

$$J=\frac{1}{2}(x-\alpha e)^T\Sigma^{-1}(x-\alpha e)+\lambda\left(\sum_{i=1}^{p}\alpha_{\mathrm{i}}-1\right). \tag{2.91}$$

Abundance estimation is obtained by taking the derivation of energy function with respect to α and equating the result to zero:

$$\begin{gathered}\frac{\partial J}{\partial\alpha}=-e^T\Sigma^{-1}x+\hat{\alpha}e^T\Sigma^{-1}e+\lambda=0\Rightarrow\\ \hat{\alpha}=(e^T\Sigma^{-1}e)^{-1}e^T\Sigma^{-1}x-(e^T\Sigma^{-1}e)^{-1}\lambda\\ \lambda=e^T\Sigma^{-1}(x-\hat{\alpha}e)\end{gathered} \tag{2.92}$$

Applying GLRT with these estimated parameters, HSD algorithm can be obtained as

$$T_{\text{HSD}}(x)=\frac{(x-\hat{\alpha}_{\mathrm{b}}b)^T\Sigma^{-1}(x-\hat{\alpha}_{\mathrm{b}}b)}{(x-\hat{\alpha}e)^T\Sigma^{-1}(x-\hat{\alpha}e)}. \tag{2.93}$$

2.4.3 Hybrid Unstructured Detector (HUD)

The HUD algorithm applies multivariate Gaussian distribution to model the background like ACE algorithm and also performs whitened FCLS counterpart for the abundance estimation [12]. In ACE algorithm, the abundance estimation is found as shown in Equation 2.35:

$$\hat{\alpha}=(s^T\Sigma^{-1}s)^{-1}s^T\Sigma^{-1}x. \tag{2.94}$$

The ACE algorithm can be rewritten by inserting the whitened FCLS counterpart of this abundance estimation to ACE formulation which is shown in Equation 2.40. Therefore, the HUD algorithm can be obtained:

$$T_{\text{ACE}}(x)=\frac{x^T\Sigma^{-1}s(s^T\Sigma^{-1}s)^{-1}s^T\Sigma^{-1}x}{x^T\Sigma^{-1}x}\Rightarrow T_{\text{HUD}}(x)=\frac{x^T\Sigma^{-1}s\hat{\alpha}}{x^T\Sigma^{-1}x} \tag{2.95}$$

In this chapter, generally the target detection methods in the literature are given. In this thesis, these methods are applied to datasets and compared in terms of performance metrics.

CHAPTER 3

SUPERPIXEL EXTRACTION METHODS

Superpixel segmentation is used to group pixels which have similar features. Using perceptually meaningful segmented regions as inputs for analysis provides advantages in the decreasing of the complexity and the elimination of redundant data. Therefore, faster and more efficient results can be obtained by using superpixels. The main expectations from a superpixel algorithm can be summarized as:

1. Boundaries of the image should be the boundary pixels of segmented superpixels.

2. Superpixel algorithm should be fast, memory efficient, and simple to use.

3. The performance of the overall image segmentation should be high.

Although there are a lot of algorithms for superpixels in the literature, the best known algorithm is Simple Linear Iterative Clustering (SLIC) algorithm [14]. SLIC algorithm has many advantages such as good segmentation performance, easy to understand and open source coding. Another efficient superpixel algorithm is boundary update based approach [16]. This algorithm is proven to be successful of extracting superpixels efficiently by updating boundaries after each iteration. These algorithms and their advantages/disadvantages are explained in detail in this chapter. Although these algorithms are developed for RGB images, they are adapted to hyperspectral images. These algorithms are basis for the proposed method in this thesis.

3.1 Simple Linear Iterative Clustering (SLIC)

SLIC algorithm divides image into equal sized regions at first. These regions are updated by using specific metrics iteratively to adhere to image boundaries. As a result, segmented regions can be utilized for classification.

SLIC algorithm initially creates the equal sized square regions depending on the parameter *k* which is the number of superpixels. Initial segmentation is extremely important because it affects the average size of superpixels. The size of the initial clusters is found by the equation $S = \sqrt{N/k}$ where *N* is the number of all pixels in the image and *S* can be defined as the regular grid interval between the superpixels. Another initial operation in the superpixelization process is transformation of color images from RGB to LAB space which is a color space defined by International Commission on Illumination (CIE). Cluster centers can be represented as $C_{\text{i}} = [l_{\text{i}}\, a_{\text{i}}\, b_{\text{i}}\, x_{\text{i}}\, y_{\text{i}}]^T$ where l, a, b are the mean color space values of superpixels and x, y represent the mean position of the cluster centers. At the end of the initialization process, to prevent cluster centers placed on the edge or noisy pixel, these centers should be moved to the lowest gradient position within a 3x3 neighborhood.

To add a pixel to a superpixel region the distances between each pixel and the nearest cluster centers are calculated in a limited search region which has the size of $2S$x$2S$. Two different distance measurements which are color distance d_{c} and spatial distance d_{s} are applied for each pixel. These distance metrics are combined by normalizing their maximum distances within a cluster and the result is represented as D'. The formulations for all distance metrics are given below.

$$\begin{aligned} d_{\text{c}} &= \sqrt{(l_{\text{j}} - l_{\text{i}})^2 + (a_{\text{j}} - a_{\text{i}})^2 + (b_{\text{j}} - b_{\text{i}})^2} \\ d_{\text{s}} &= \sqrt{(x_{\text{j}} - x_{\text{i}})^2 + (y_{\text{j}} - y_{\text{i}})^2} \\ D' &= \sqrt{\left(\frac{d_{\text{c}}}{N_{\text{c}}}\right)^2 + \left(\frac{d_{\text{s}}}{N_{\text{s}}}\right)^2} \end{aligned} \tag{3.1}$$

The maximum color distance and spatial distance are represented with two parameters denoted as N_{c} and N_{s} respectively. The maximum value of N_{s} equals to S in a cluster. Therefore, N_{s} can be replaced by S in the formulation. However, to determine the value of N_{c} is not easy due to the different color distance values of each cluster. For this reason, a constant value m is used instead of N_c. As a result, the overall distance metric can be formulated as

$$D = \sqrt{{d_\mathrm{c}}^2 + \left(\frac{d_\mathrm{s}}{S}\right)^2 m^2}\,. \tag{3.2}$$

m can be thought of as the weight parameter which strongly depends on the determined priority of spatial or color distances by the user. If m is large, then the spatial distance has more importance and the shape of the superpixels is more regular. Otherwise, image boundaries are coherent in resulting superpixels and the shape of the superpixels is less regular.

After distance metrics are computed, the pixel is assigned to the cluster center which has the lowest distance with that pixel. Then, the cluster centers and mean values of the superpixels which can be represented as the vector $[l\,a\,b\,x\,y]^T$ are updated. Residual errors are computed by calculating L_2 distance between the updated cluster centers and the previous cluster centers. This procedure, which includes assignment and update steps, is repeated until the residual error converges to the predetermined threshold value. However, iteration number is determined as 10 due to the satisfactory results for most images [14]. Some of the pixels can be disjoint from their assigned superpixels after applying this procedure. These pixels are reassigned to the nearest superpixels as a post-processing step to provide connectivity.

SLIC algorithm is summarized below [14]:

- Initialization Process:
    - Calculate the parameters represented as $C_\mathrm{k} = [l_\mathrm{k}\,a_\mathrm{k}\,b_\mathrm{k}\,x_\mathrm{k}\,y_\mathrm{k}]^T$ for all cluster centers which are initially sampled at S pixels.
    - Find the lowest gradient positions within a 3x3 neighborhood of cluster centers and move the cluster centers to these found positions.
    - Assign -1 value to the label of each pixel ($l(i) = -1$).
    - Assign ∞ to the distance of each pixel ($d(i) = \infty$).

  Repeat{

- Assignment Process:

```
for each cluster center $C_k$ do
    for each pixel $i$ in a $2Sx2S$ region around $C_k$ do
        Calculate the distance $D$ between $C_k$ and $i$.
        if $D < d(i)$ then
            set $d(i) = D$
            set $l(i) = k$
        end if
    end for
end for
```

- Update Process:

```
Compute new cluster centers.
Compute the residual error $E$.
} until $E \leq threshold$
```

Figure 3.1: RGB image and SLIC superpixels on RGB image for different m values

Figure 3.1 illustrates an example of the implementation of the SLIC algorithm with different m values. The regularity of the superpixels depends on the m parameter.

3.2 Superpixel Extraction Method Based on Boundary Update

The main disadvantages of the SLIC algorithm are the memory inefficiency and computational cost due to a large number of pixels in $2Sx2S$ area for every superpixel in each iteration. To overcome these problems, boundary update based approach is proposed [16].

The initialization process in this algorithm is the similar to the SLIC algorithm. After regular grids are created, 8-neighborhood search is applied to the boundary pixels to find the neighbor superpixels. The distance metrics between boundary pixels of the superpixels and these neighbor superpixel centers are calculated. As in the SLIC algorithm, color distance and spatial distance are used for the update of the boundaries. The formulation of the color distance is shown as

$$C(p, Q) = \sum_{i=1}^{3} (p^i - Q^i)^2 \tag{3.3}$$

where p is the boundary pixel, Q is the mean color value of neighbor superpixel and i represents the channel which has the color information. Both of RGB and LAB color spaces can be chosen. For the spatial distance, Euclidean distance and geodesic distance are considered. The Euclidean distance is given as

$$D(p, Q^C)_{\text{EUCLIDEAN}} = \sqrt{(p_{\text{x}} - {Q_{\text{x}}}^C)^2 + (p_{\text{y}} - {Q_{\text{y}}}^C)^2}, \tag{3.4}$$

where Q^C represents the cluster center and ${Q_{\text{x}}}^C$, ${Q_{\text{y}}}^C$ are the position of the cluster center.

Geodesic distance is the path which has minimum cost of all possible paths between the pixel and superpixel center. The path length $l(P)$, which is defined as the sum of distances between neighbor pixels represented as $d_{\text{N}}(p_{\text{i}}, p_{\text{i+1}})$, is formulated as

$$l(P) = \sum_{i=1}^{n-1} d_{\text{N}}(p_{\text{i}}, p_{\text{i+1}}). \tag{3.5}$$

The distance between neighbor pixels d_{N} can be calculated as

$$d_{\mathrm{N}}(p,q)=\sum_{i=1}^{3}(1+|p_{\mathrm{i}}-q_{\mathrm{i}}|). \tag{3.6}$$

RGB or LAB color spaces, which have three color channels, can be chosen to calculate d_{N}.

The geodesic distance can be found by finding the minimum of all distances:

$$D(p,Q^{C})_{\mathrm{GEODESIC}}=\min_{P=p_1,p_2,\dots p_{\mathrm{n}}} l(P) \tag{3.7}$$

To update boundaries, an energy function which is a combination of color and spatial distances is used. Energy function can be formulated as

$$E(p,Q)=\lambda C(p,Q)+(1-\lambda)D(p,Q^{C}) \tag{3.8}$$

where λ is the convexity weight which affects the result of superpixel algorithm in terms of shape regularity and segmentation.

In the proposed superpixel extraction algorithm, boundary pixels are assigned to the superpixels which have the minimum energy value in the boundary update process. Mean values of the new superpixels, whose boundary pixels are changed, are computed in the structure update process. Boundary update and structure update processes are applied iteratively as in the SLIC algorithm. The iteration number is experimentally selected as 10 due to the sufficient performance. As a post-processing step, a median filter is applied to the pixels to avoid the connectivity problem within the superpixels.

Figure 3.2 illustrates the superpixel extraction results of boundary update based method for different λ values and the same number of superpixels. Like the m parameter in the SLIC algorithm, λ values affect regularity.

In Figure 3.3, the updated superpixel regions of iterations are given. From the first iteration to the last iteration the boundaries of the superpixels are getting closer to object boundaries.

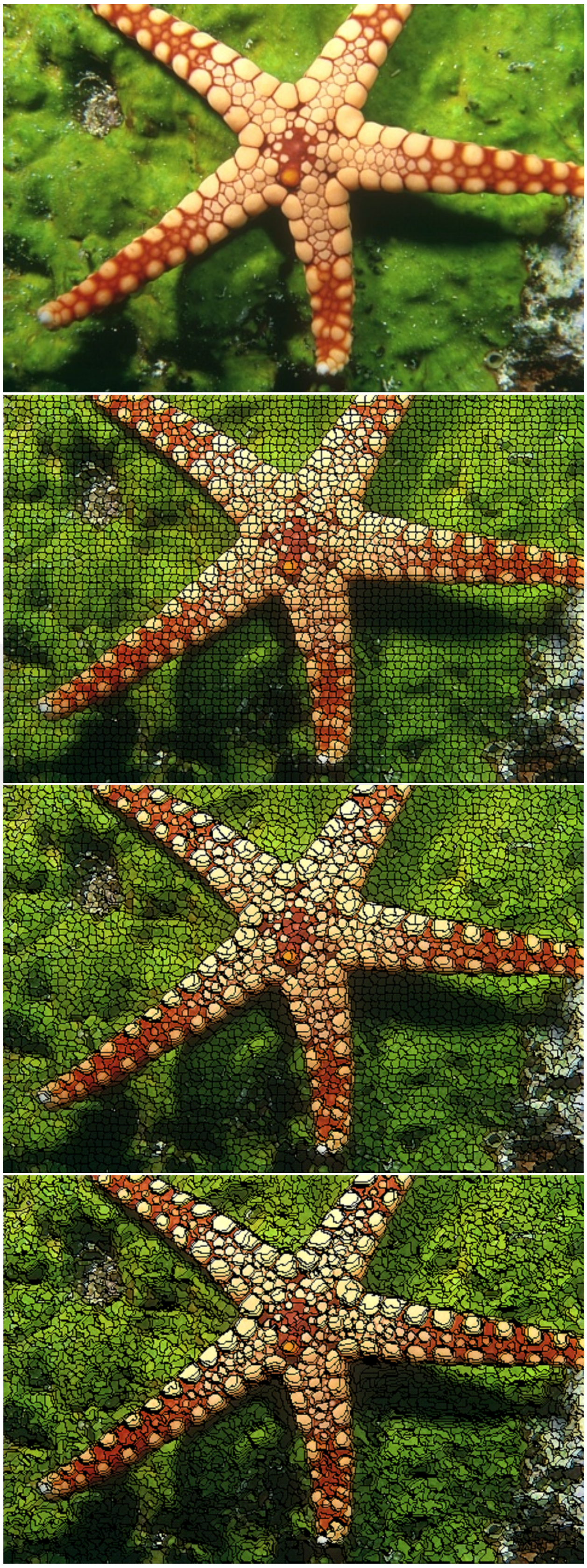

Figure 3.2: RGB image and superpixels which are created by boundary update-based method for λ=[0.15, 0.5, 1] values

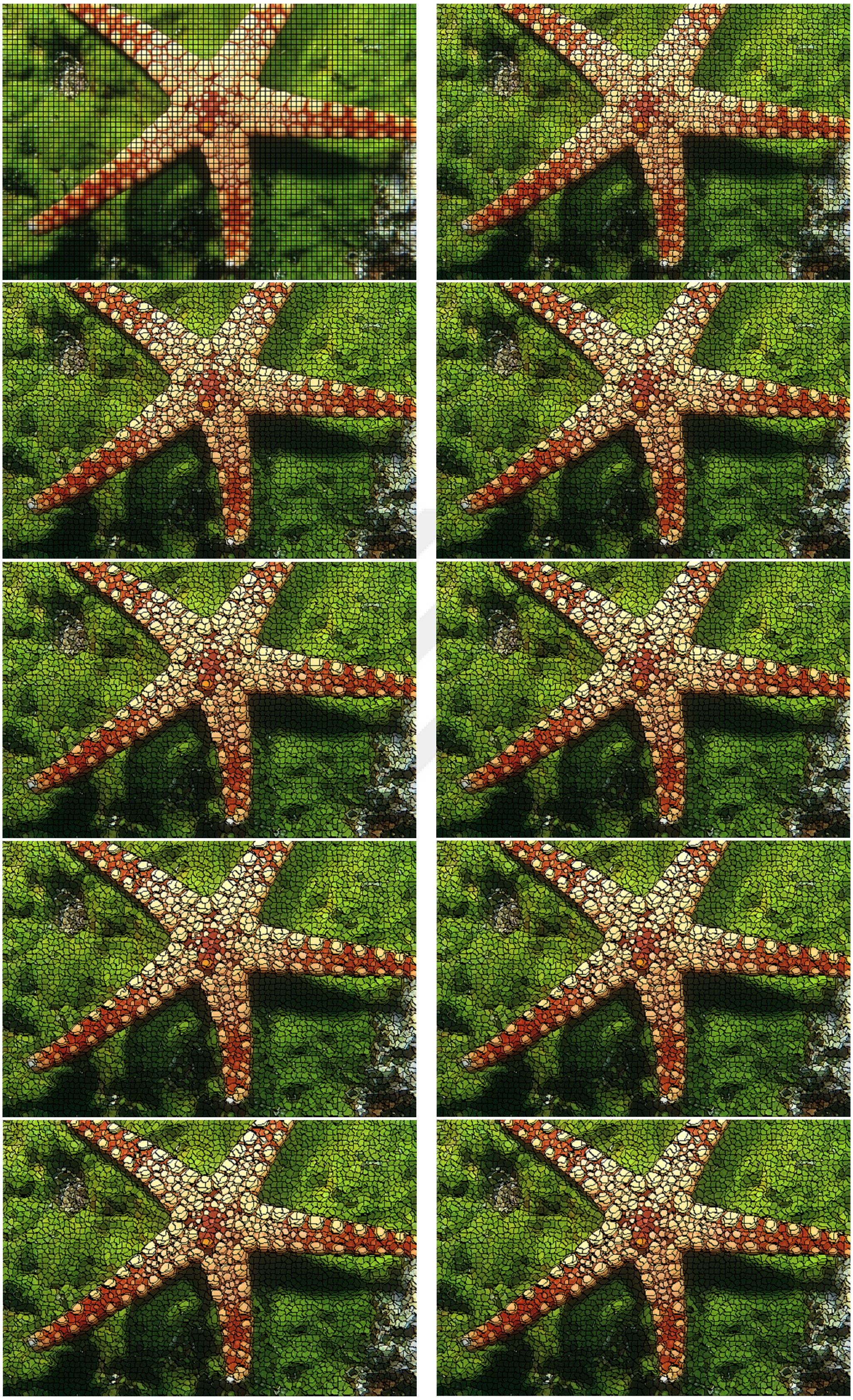

Figure 3.3: The superpixel results of boundary update based superpixel extraction method after each iteration

CHAPTER 4

PROPOSED SUPERPIXEL BASED TARGET DETECTION METHODS

In this chapter, the proposed superpixel-based target detection methods are explained in detail. Firstly, adaptation of the superpixel algorithms to the hyperspectral images is applied. After the superpixels of hyperspectral images are created, the superpixel representatives which provides the spectral characteristic of the superpixel regions are obtained to use for background modeling, matching or abundance estimation depending on the target detection algorithms. Applying the target detection algorithms to these superpixels is the main consideration for this thesis. Different representatives of superpixels such as mean, medoid, and centroid can be used for analysis of the proposed target detection algorithms. Superpixel representatives not only show the spectral characteristics of the corresponding region, but also they are analyzed faster than all image pixels. Figure 4.1 illustrates the main steps which are mentioned in this chapter.

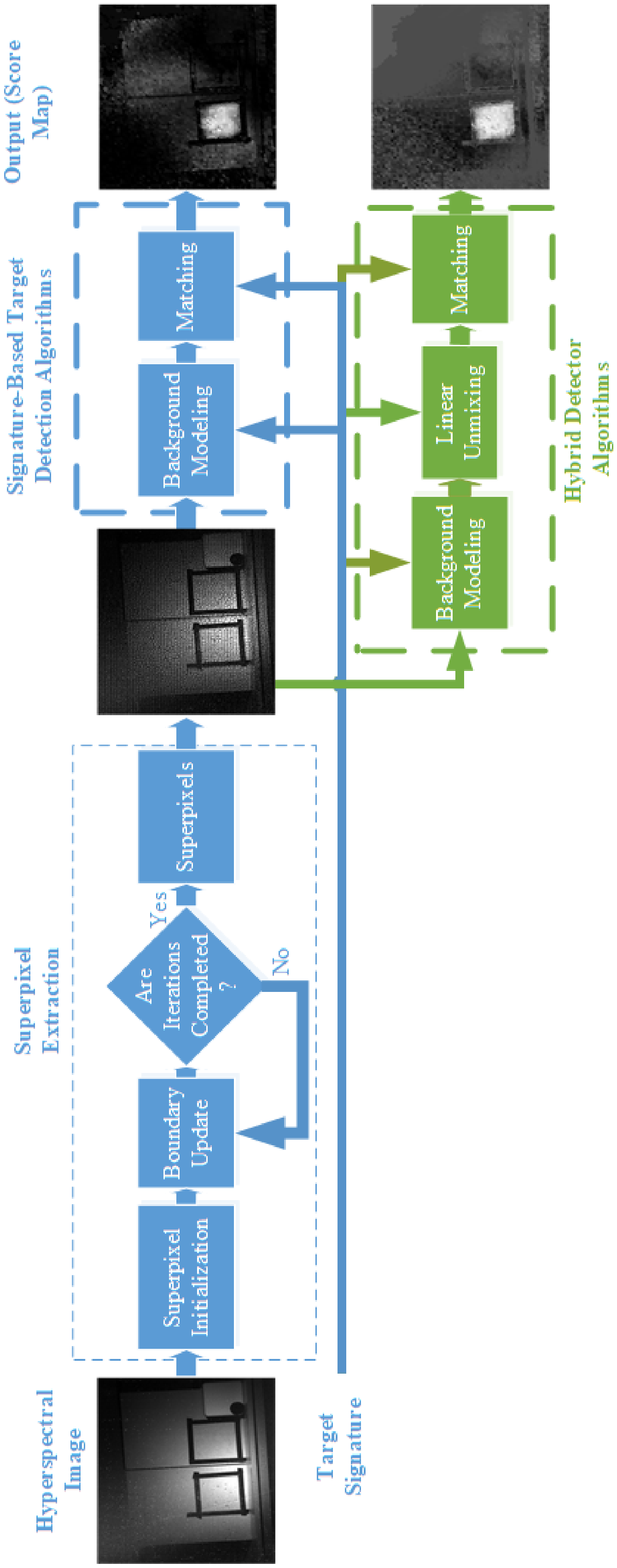

Figure 4.1: Block diagram of the proposed system

4.1 Adaptation of Superpixel Extraction Method to Hyperspectral Images

The boundary update-based superpixel algorithm and SLIC algorithm can be applied to the hyperspectral images. In this chapter, adaptations of the SLIC algorithm and the boundary update-based superpixel extraction method to hyperspectral images are explained [18].

As the first approach, the SLIC algorithm is adapted to the hyperspectral images by simply changing the band number in the MSE calculation for the spectral distance. The main difference between RGB and hyperspectral images is the number of spectral bands in terms of extraction of the superpixels. Euclidean distance can be found by using the formulation given in Equation 3.1. Because hyperspectral images have more than three bands, this equation can be rewritten as

$$d_{\text{spectral}} = \sqrt{\sum_{i=1}^{l}(x_{\text{i}} - y_{\text{i}})^2} \tag{4.1}$$

where x is the boundary pixel vector, y is the cluster center spectrum and l is the number of spectral bands.

The procedure of the boundary update-based superpixel extraction method is the same with RGB superpixel extraction mentioned in Section 3.2 except the calculation of the spectral distance. The various spectral distance metrics such as Euclidean distance, SAM, and Spectral Information Divergence (SID) can be applied to the hyperspectral images for obtaining the superpixels [18]. Euclidean distance can be calculated by using Equation 4.1.

SAM gives the angle between two vectors which can be used as a spectral distance metric. Its formulation can be given as

$$T_{\text{SAM}}(x) = arccos\left(\frac{s^T x}{(s^T s)^{\frac{1}{2}}(x^T x)^{\frac{1}{2}}}\right). \tag{4.2}$$

Another spectral distance metric is SID which employs probabilistic approach by using the elements of the pixel vector. SID is suggested metric for hyperspectral images instead of using SAM due to its effective results for characterizing the spectral variability [33]. Reflectance or radiance values can be shown as a pixel vector

$x = (x_1, ..., x_\mathrm{L})^T$. The elements of this vector are assumed as non-negative numbers. Therefore, a probability measure can be defined as

$$P(\lambda_\mathrm{j}) = p_\mathrm{j} = \frac{x_\mathrm{j}}{\sum_{i=1}^{L} x_\mathrm{i}} \tag{4.3}$$

where L represents the number of bands of the hyperspectral image. The desired probability vector $\mathbf{p} = (p_1, p_2, ...p_\mathrm{L})^T$ describes the spectral variability of a pixel.

Another pixel and its probability vector can be shown as $y = (y_1, ..., y_\mathrm{L})^T$ and $\mathbf{q} = (q_1, q_2, ...q_\mathrm{L})^T$ respectively. ith band self-information of two different pixels can be shown as $I_\mathrm{i}(x) = -logp_\mathrm{i}$ and $I_\mathrm{i}(y) = -logq_\mathrm{i}$. The relative entropy or Kullback-Leibler divergence of **y** can be formulated as

$$D(x||y) = \sum_{i=1}^{L} p_\mathrm{i} D_\mathrm{i}(x||y) = \sum_{i=1}^{L} p_\mathrm{i}(I_\mathrm{i}(y) - I_\mathrm{i}(x)) = \sum_{i=1}^{L} p_\mathrm{i} log\left(\frac{p_\mathrm{i}}{q_\mathrm{i}}\right). \tag{4.4}$$

The similarity between two pixels can be measured by using SID which can be defined as

$$SID(x, y) = D(x||y) + D(y||x). \tag{4.5}$$

Boundary update-based superpixel extraction can be applied to hyperspectral images by using one of these three distance metrics for spectral distance. Computation of spatial distance is the same with RGB images for both superpixel extraction methods. Figure 4.2 shows the boundary update-based superpixel extraction method results for Indian Pines dataset [1] with the ground truth of classified regions. As it can be seen from the figure, the boundaries of the superpixels are compatible with the ground truth of this dataset.

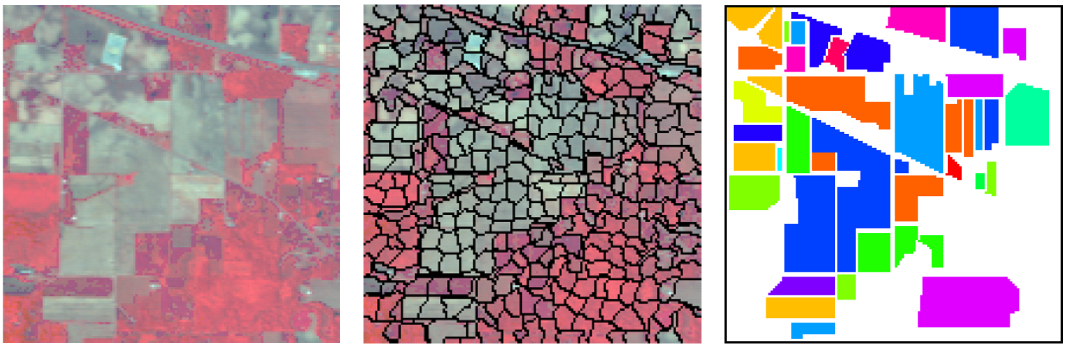

Figure 4.2: a) RGB image of Indian Pines dataset b) obtained superpixel results by using SAM as the spectral distance metric and c) ground truth of Indian Pines dataset

4.2 Proposed Superpixel-Based Target Detection Methods

After the superpixel extraction process is completed, the superpixel representatives are given as inputs to the target detection algorithms. These representatives are utilized in background modeling, matching or abundance estimation depending on the target detection algorithm. Superpixel-based target detection algorithms enhance the detection performance not only by eliminating noise effects at the result but also by taking spatial relations between neighbor pixels into consideration. Signature-based target detection algorithms contain two parts which are denoted as background modeling and matching. Background modeling and matching can be defined as applying covariance matrix that is used to increase the detection scores of pixels which have the high correlation with target spectrum and to decrease the scores of other pixels.

There are different combinations in terms of background modeling and matching to apply superpixel-based target detection algorithms to hyperspectral images [34]. These combinations are summarized in Table 4.1. The main aim is applying the combinations to hyperspectral images and comparing the detection performance of the results. The results are discussed and commented in Section 6.2.

Background Modeling	**Matching**
Pixels	Pixels
Pixels	SP Representatives
SP Representatives	Pixels
SP Representatives	SP Representatives

Table 4.1: The possible combinations of pixels and superpixel representatives for background modeling and matching in superpixel based target detection algorithms

Hybrid target detection algorithms can be adapted to superpixels by abundance estimation as an additional part to background modeling and matching. Abundance estimation of target expands the combinations and provides to add hyperspectral unmixing as a useful feature of the proposed system. All of the possible combinations are given in Table 4.2.

Background Modeling	Matching	Abundance Estimation
Pixels	Pixels	Pixels
Pixels	Pixels	SP Representatives
Pixels	SP Representatives	Pixels
Pixels	SP Representatives	SP Representatives
SP Representatives	Pixels	Pixels
SP Representatives	Pixels	SP Representatives
SP Representatives	SP Representatives	Pixels
SP Representatives	SP Representatives	SP Representatives

Table 4.2: The possible combinations of pixels and superpixel representatives for background modeling, matching, and abundance estimation in superpixel-based hybrid detector algorithms

The selection of superpixel representatives affects the detection results, therefore the determination of the superpixel representatives is one of the main issues for superpixel-based target detection. Three different methods which are mean, medoid and centroid were proposed to determine superpixel representatives [34]. The mean of a superpixel can be found simply by taking the average reflectance or radiance value at each wavelength of all pixels in the superpixel. The formulation of the mean of the kth spectral band of the ith superpixel can be given as

$$\mu_{\mathrm{k}}(i) = \frac{1}{N}\sum_{j=1}^{N} x_{\mathrm{jk}} \quad k = 1, ..., l \tag{4.6}$$

where N is the number of pixels in the superpixel, x_{jk} is the reflectance or radiance value of jth pixel at kth band and l is the number of wavelengths. After obtaining mean reflectance or radiance value at each band, the results can be concatenated to form the mean spectrum of the superpixel. Although choosing mean spectrum as the representative is very useful to decrease the effect of noisy pixels in the superpixel region, the size of the superpixel should be much greater than the number of noisy pixels. As the number of superpixel increases, the selection of mean as the representative is more risky due to the occurrence of the smaller size of superpixels.

As the second representative, medoid can be defined as the closest pixel to all other pixels in the superpixel. To find the medoid of a superpixel, Euclidean distances of each pixel to other pixels is computed and the summation of these distances is compared with total distances of other pixels. The pixel which has the minimum distances to other pixels is the medoid of the superpixel. The formulation of the total Euclidean distances for mth pixel can be given as

$$d_{\text{euclidean}}(m) = \sum_{\substack{j=1 \\ m \neq j}}^{N} \sum_{k=1}^{l} \sqrt{(x_{\text{mk}} - x_{\text{jk}})^2} \qquad m = 1, ..., N. \tag{4.7}$$

Assuming tth pixel has the minimum Euclidean distance, then the medoid of the ith superpixel can be given as

$$\begin{aligned} d_{\text{euclidean}}(t) &= \min_{d_{\text{euclidean}}} \left(d_{\text{euclidean}}(m) \right) \quad m = 1, ..., N \\ medoid(i) &= x_{\text{t}}. \end{aligned} \tag{4.8}$$

The satisfactory results can be obtained by choosing medoid as the superpixel representative. A simple example for the comparison between the mean and medoid in the case of an existing outlier is given in the Figure 4.3 [35]. As it can be understood from this figure, the selection of medoid is more reliable than the selection of mean as the representative. Another important difference between medoid and mean is that the medoid is the element of the set, and not a synthetically generated sample.

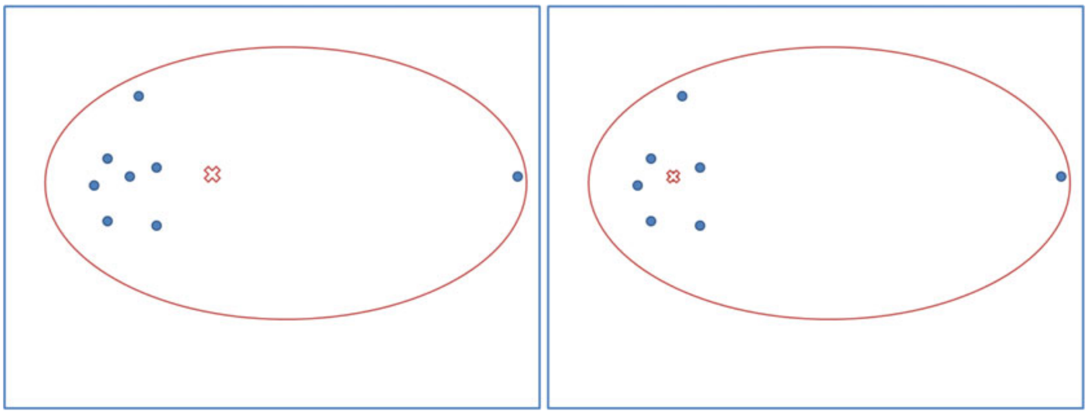

Figure 4.3: a) Mean of the set b) Medoid of the set

As the last alternative among the superpixel representatives, centroid is the pixel which is at the center position of the superpixel. Center position of the ith superpixel can be found as

$$\begin{aligned} x_{\text{center}}(i) &= \frac{1}{N}\sum_{j=1}^{N} x_{\text{j}} \\ y_{\text{center}}(i) &= \frac{1}{N}\sum_{j=1}^{N} y_{\text{j}} \end{aligned} \tag{4.9}$$

where x and y are the horizontal and vertical positions of the pixels, N is the number of pixels in the superpixel. The centroid of the ith superpixel can be given as

$$centroid(i) = \mathbf{z}(x_{\text{center}}(i), y_{\text{center}}(i)) \tag{4.10}$$

where **z** is the pixel vector which is located at the nearest position to ($x_{\text{center}}(i)$, $y_{\text{center}}(i)$).

As an illustrative example, Figure 4.4 gives the reflectance vs wavelength graphs for the representatives of a sample superpixel. Although there are small differences between the representatives as it can be seen from the graphs, the detection results could be very different for the superpixels which includes target spectrum. In this thesis study, all of these superpixel representatives are compared in terms of the detection performance and the results are given in Section 6.2.

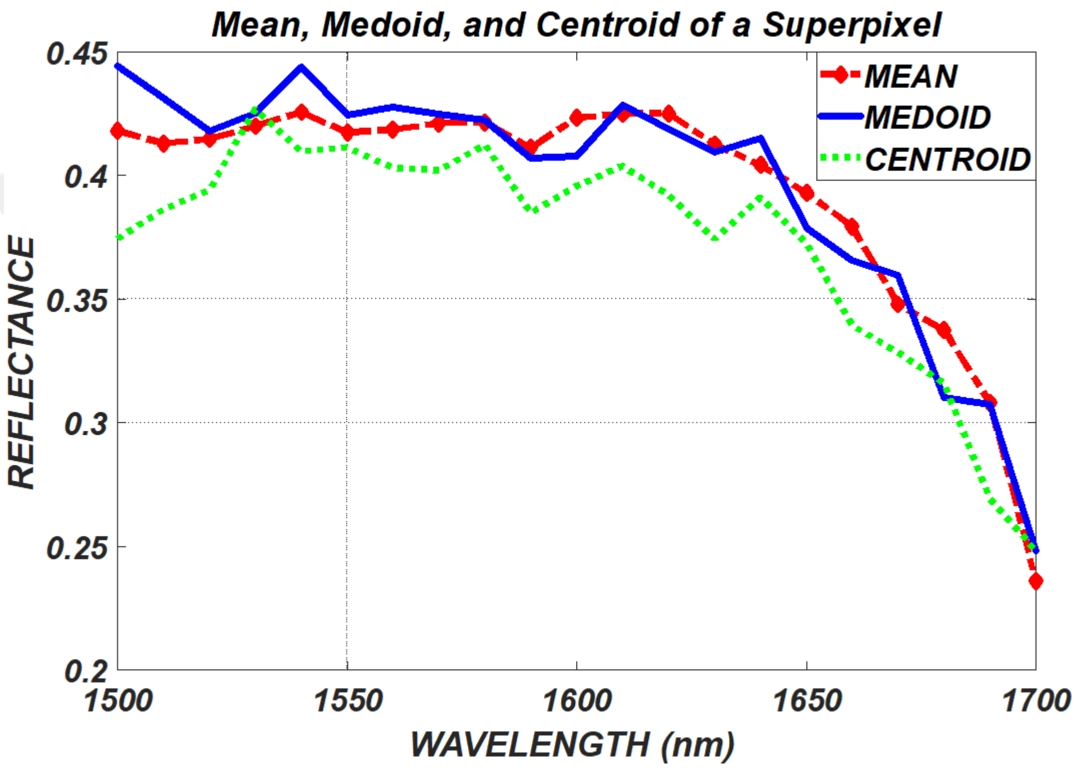

Figure 4.4: An illustrative example of reflectance-wavelength graphs for mean, medoid, and centroid of a sample superpixel

CHAPTER 5

EXPERIMENTAL SETUPS AND PERFORMANCE METRICS

Hyperspectral image datasets are rarely found on the internet as mentioned before. Most of them are not appropriate for the target detection due to the lack of target spectrum. Therefore, an experimental setup is prepared to capture hyperspectral images along with the spectral signatures of the determined target materials. In particular, a SWIR broadband camera with a Liquid Crystal Tunable Filter (LCTF) is used for capturing. An ASD spectroradiometer is utilized to obtain target spectra library. In this chapter, the experimental setup which include these instruments are given in detail.

Although radiance values could also be used for the aim of target detection, reflectance values are used as inputs for the signature-based target detection algorithms to obtain a better discrimination between background and target materials. Therefore, the utilized methodology for radiance to reflectance conversion is also clarified in this chapter.

Finally, the performance metrics which are utilized to compare the performance of different algorithms for superpixel extraction and target detection are presented and explained in this chapter.

5.1 Instruments and Experimental Setups

To capture hyperspectral images, a SWIR hyperspectral camera and an LCTF are used. The utilized SWIR hyperspectral camera, which is produced by Mikro-Tasarım

Company, has the following features [36]:

- capable of capturing between 1000-1700 nm spectral bands,
- frame rate up to 70 frames per second (fps),
- 640x512 resolution.

The SWIR band is specially selected due to the existing of discriminative characteristics of target materials in this interval.

LCTF is a hardware that is used to tune the spectral wavelength within the determined time interval. VariSpec LNIR, which is produced by Perkin Elmer Company, is used as an LCTF model for this study. The features of the LCTF are summarized as [37]:

- spectral range is 850-1800 nm,
- bandwidth options are 6 and 20 nm,
- 150 ms response time.

Figure 5.1 illustrates the block diagram of the developed hyperspectral imaging system for the experiments.

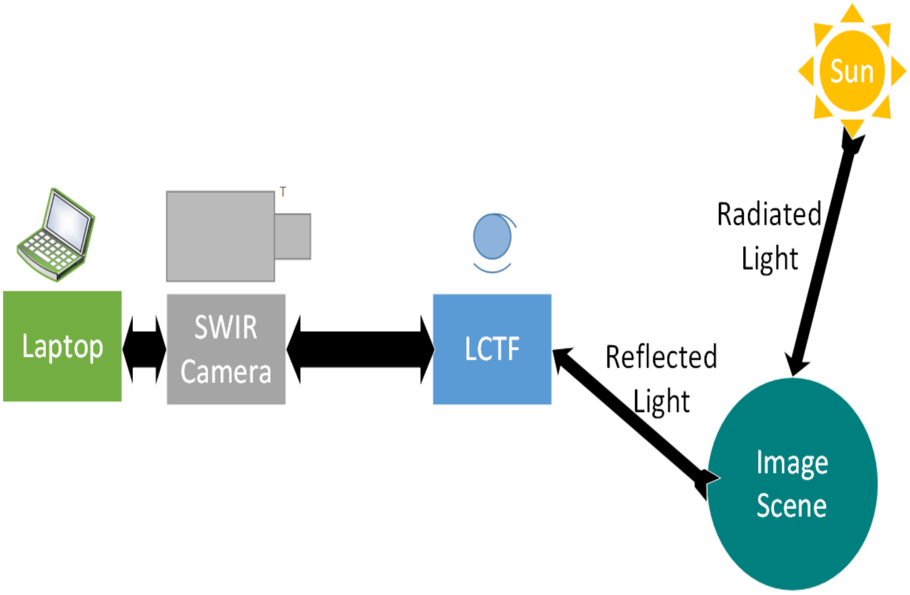

Figure 5.1: Block diagram of the hyperspectral imaging system

The utilized hyperspectral SWIR camera and LCTF are shown in Figure 5.2 and Figure 5.3 respectively:

Figure 5.2: Mikro-Tasarım SWIR Hyperspectral Camera

Figure 5.3: Liquid Crystal Tunable Filter (LCTF)

During the capturing process, LCTF is placed in front of the SWIR camera as shown in the Figure 5.4, to collect the lights at the determined wavelengths:

Figure 5.4: SWIR camera and LCTF

Another utilized measurement tool in the experiments is ASD Field Spectroradiometer produced by Malvern Panalytical Company. It is used to obtain the spectrum of the target material in the range 350-2500 nm. ASD spectroradiometer, which is shown in Figure 5.5, is used to create a spectral library which can be assumed as the ground truth of the target spectra. Some of the important features of ASD spectroradiometer are summarized as [38]:

- spectral range is 350-2500 nm,
- scanning time is 100 ms,
- channel number is 2151.

Figure 5.5: ASD spectroradiometer

There are different measurement tools of ASD spectroradiometer for the application areas and these tools are connected ASD spectroradiometer with a fiber optic cable. The contact probe, which is a measurement tool that is contacted to the material, can be used to obtain the most accurate spectrum of the material. The contact probe which is shown in Figure 5.6 is designed to measure the spectra of solid materials. It uses a halogen bulb as a light source.

Another measurement tool is the pistol grip which is shown in Figure 5.6. It provides the possibility of obtaining the spectrum without any contact. The target spectrum which is obtained by the pistol grip has some noisy bands and lower amplitudes than the original spectrum. However, it is very useful for the spectral measurement of explosive materials which are sensitive to light and physical impacts. On the other hand, measurement with pistol grip requires the external light source which should provide light in the SWIR range.

Figure 5.6: a) Contact probe of ASD spectroradiometer b) pistol grip of ASD spectroradiometer

The calibration of the ASD spectroradiometer is the most important operation for accurate measurement of the spectra. A spectralon (Figure 5.7), which is used to calibrate the spectroradiometer, reflects almost all of the incoming lights with its characteristic. Before the measurement of the target material, the contact probe or pistol grip is used to measure the reflectance of the spectralon. The reflectance of the spectralon is used as the reference spectrum for the light and the reflectance of the target material is measured according to this reference spectrum.

Figure 5.7: Spectralon

5.2 Datasets

The hyperspectral image datasets on the internet include hyperspectral images which are captured by an airplane or a satellite. Some of these datasets such as Indian Pines and Salinas are used for testing of superpixel algorithms because their ground truth images are provided with these datasets [1].

Indian Pines includes 200 spectral bands, which are the remaining bands after eliminating some distorted bands due to atmospheric effects. The size of this dataset is 145x145 and the captured data is in the range of 400-2500 nm. A sample band and the ground truth of Indian Pines are given in Figure 5.8. The ground truth classes of this dataset and their respective samples number are given in Table 5.1. There are 16 different classes in the ground truth.

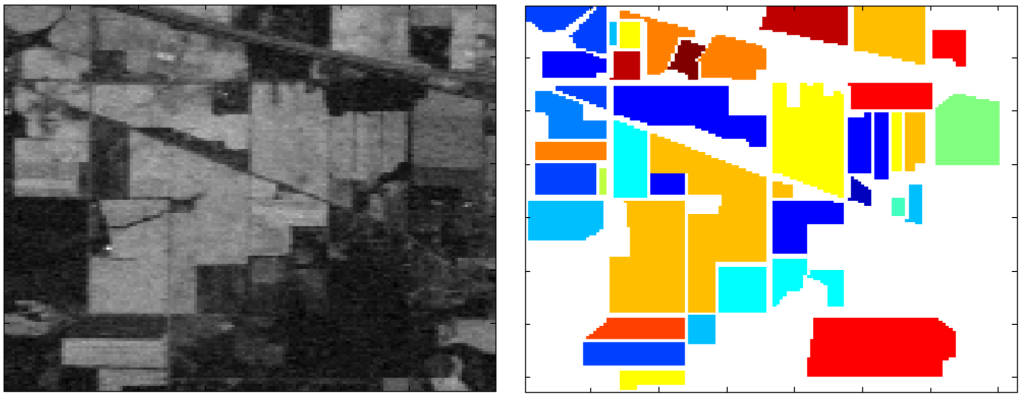

Figure 5.8: a) Sample band of Indian Pines dataset b) ground truth of Indian Pines dataset [1]

Class Number	Class	Sample Number
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Building-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table 5.1: The ground truth classes of Indian Pines dataset [1]

Salinas contains 204 spectral bands and 512x217 pixels in the range of 400-2500 nm. A sample band and the ground truth of Salinas are given in Figure 5.9. Salinas dataset includes 16 classes which are shown in the Table 5.2.

The ground truth information is very precious, because most of the datasets do not include this information. On the other hand, Indian Pines and Salinas datasets do not include the spectrum of the ground truth classes. Therefore, these datasets could not

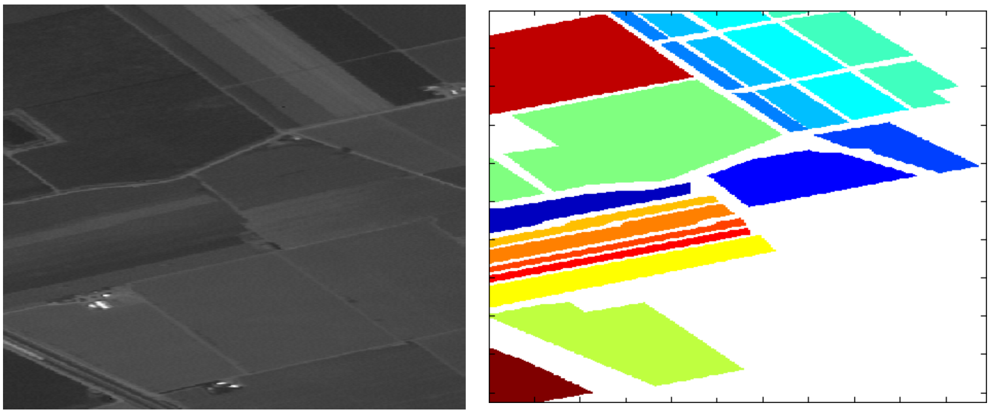

Figure 5.9: a) Sample band of Salinas dataset b) ground truth of Salinas dataset

Class Number	Class	Sample Number
1	Brocoli-green-weeds-1	2009
2	Brocoli-green-weeds-2	3726
3	Fallow	1976
4	Fallow-rough-plow	1394
5	Fallow-smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes-untrained	11271
9	Soil-vinyard-develop	6203
10	Corn-senesced-green-weeds	3278
11	Lettuce-romaine-4wk	1068
12	Lettuce-romaine-5wk	1927
13	Lettuce-romaine-6wk	916
14	Lettuce-romaine-7wk	1070
15	Vinyard-untrained	7268
16	Vinyard-vertical-trellis	1807

Table 5.2: The ground truth classes of Salinas dataset [1]

be used for signature-based target detection algorithms. Assuming the same class pixels have the similar spectral signatures, testing superpixel algorithms on these datasets is possible.

Digital Imaging and Remote Sensing group of Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology (RIT) shares a dataset which is captured for the target detection purposes [39]. Spectral signatures and position information of the target materials are included in this dataset. Therefore, this dataset is appropriate for the target detection and it is used in some studies [40, 41]. RGB image of the dataset is given below.

Figure 5.10: RGB image of RIT dataset

The dataset, which is obtained by the system shown in Figure 5.1 for this study, is prepared for different conditions and purposes. Unlike other datasets, this dataset is prepared for nearer chemical substance targets such as salt and sugar. The spectral signatures of these chemical substances are obtained by using ASD spectroradiometer. On the other hand, solid and soluble forms of these substances are prepared for the experiments. This hyperspectral dataset includes solid and soluble forms of the salt compound. The aqueous salt solution which has the density of 250 gr/L is sprayed on colored metal plates homogeneously. An RGB image, a sample band and the ground truth of the dataset are given below.

Figure 5.11: RGB image of the prepared dataset for outdoor experiment

Figure 5.12: Sample band of the prepared dataset for outdoor experiment

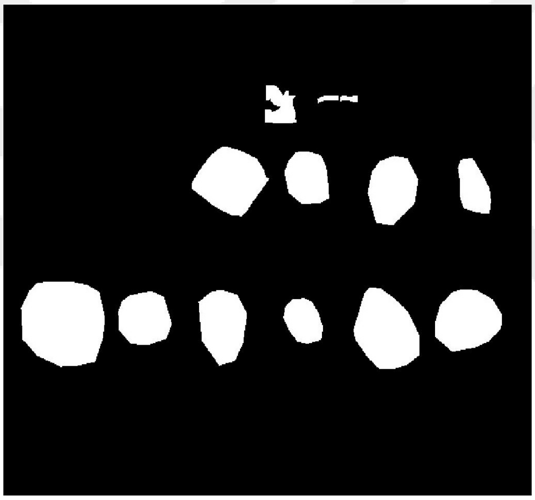

Figure 5.13: Ground truth of dataset for the solid salt compound and the aqueous salt solution

The hyperspectral image data shown in Figure 5.12 is captured outdoor on a sunny day. The sun as a light source of this experiment has the broad range of radiation. The solar spectrum and the atmospheric effects which cause distortion in some wavelengths are given and discussed in Section 6.2. The distortion in the spectral bands does not exist for indoor experiments because these experiments are applied in the close distance and atmospheric distortions can be negligible. Therefore, both of soluble salt compound and sugar are sprayed homogeneously on white metal plates and the hyperspectral image is captured for indoor experiment with a SWIR camera and an external light source which provides the light in the SWIR range, as well. The target region boundaries are set with the black tapes. The aqueous salt solution which has the density of 100 gr/L is in the left square region and the aqueous sugar solution which has the density of 80 gr/L is in the right square region. An RGB image, a sample band and the ground truth of the dataset are given below.

Figure 5.14: RGB image of the prepared dataset for indoor experiment

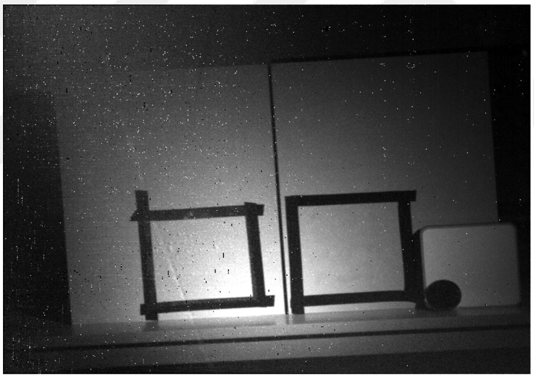

Figure 5.15: Sample band of the prepared dataset for indoor experiment

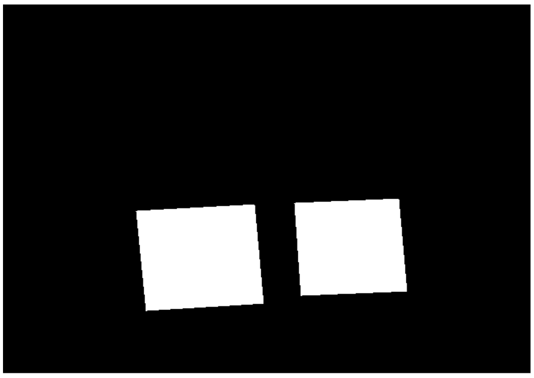

Figure 5.16: Ground truth of dataset:left-side square region includes salt and right-side square region includes sugar

5.3 Radiance-Reflectance Conversion

Reflectance is the term that defines the ratio of the amount of incoming light to the amount of leaving light from the object. Reflectance has no units. Radiance can be defined as the energy flux per solid angle leaving a unit surface area in a given direction and also it is the variable measured at the sensor. W/cm^2/sr is the unit of the radiance. Both of the radiance and reflectance data can be used for many remote sensing applications. Although they are dependent variables, reflectance variable is more reliable than radiance variable because the reflectance variable can be used to characterize the target material. Therefore, radiance-reflectance conversion is required to apply target detection algorithms to the hyperspectral images.

Spectralon shown in Figure 5.7 is the reference object which can be assumed that is the most reflective object in the scene. The dark reference object absorbs the incoming light and its reflectance value is in the range of 0.02-0.03 for all bands. The radiance-reflectance conversion can be performed by using both reference objects in the scene. The reflectance map of a hyperspectral image scene can be formulated as

$$Reflectance_\lambda = \frac{x_\lambda - d_\lambda}{w_\lambda - d_\lambda} \tag{5.1}$$

where x_λ is the sample pixel intensity at wavelength λ, d_λ is the dark reference intensity at wavelength λ and w_λ is the white reference intensity at wavelength λ [42].

The dark reference object is not an obligation in this study due to the SWIR camera properties. Although the spectral range of SWIR camera is 850-1800 nm, it performs well in the range 1000-1700 nm. There is no possibility to obtain good-quality images in 850-1000 nm and 1700-1800 nm intervals where all the pixels have the same low intensity values, which are equal to the dark reference intensity values in the 1000-1700 nm range. Therefore, any pixel can be chosen as the dark reference spectrum in the 850-1000 nm range or 1700-1800 nm range. On the other hand, spectralon object must be available on the image scene in order to use as the white reference object.

As a result, the values of the reflectance map elements are in the interval between 0 and 1. After the reflectance map is obtained, the target detection algorithms can be applied to this map by using the spectral signatures of the target materials measured by ASD spectroradiometer.

5.4 Performance Metrics

Performance metrics provide statistical results which are helpful to evaluate the results of the algorithms. Each metric is developed for evaluating algorithms accurately and tested on the ground truth observed by the humans. In this section, boundary recall and undersegmentation error which are the performance metrics for the superpixel extraction algorithms are explained. Precision and recall curves which are mostly used as performance metrics for target detection algorithms are also presented in this section. All of these metrics are widely accepted by the researchers to provide an accurate evaluation of the algorithms and to compare developed algorithms in terms of these metrics.

5.4.1 Performance Metrics for Superpixel Extraction Methods

Segmentation quality of superpixel extraction algorithms is measured by boundary recall and undersegmentation error metrics [2]. The side regions of the edges in an image mostly have different intensity values. As a requirement from the theory of superpixels, the boundaries of superpixels should trace the edges as much as possible. Boundary recall provides the rate of boundaries of superpixels which trace the segmented ground truth regions. The ground truth edges can be at most two pixels away from a superpixel boundary according to original paper [2]. The number of boundary pixels in the ground truth boundary image, which are within the range of two pixels distance to the boundary image of superpixels, is denoted as true positives. On the contrary, the number of boundary pixels in the ground truth boundary image which are not in this range to the boundary image of superpixels is called as false negatives. The boundary recall can be formulated as

$$Boundary\,Recall = \frac{True\,Positives}{True\,Positives + False\,Negatives}. \tag{5.2}$$

As the ground truth segmentation depends on the observations of humans, combining multiple ground truth boundary images by using OR operation and using this combination result as the ground truth boundary image reveals more accurate results.

Boundary recall alone is not enough to express the quality of superpixel extraction algorithms. For the case that a segment has the tortuous boundary structure whose pixels are assigned as boundary pixels, the boundary recall value would be high. Therefore, undersegmentation error is another metric that is used for evaluation of the superpixel extraction algorithm performance. It is used to measure the areas which flood over the ground truth segment borders. A superpixel is divided into two parts by a ground truth segment. These parts are called as *in* and *out* parts. Figure 5.17 illustrates these parts which are used to measure undersegmentation error [2].

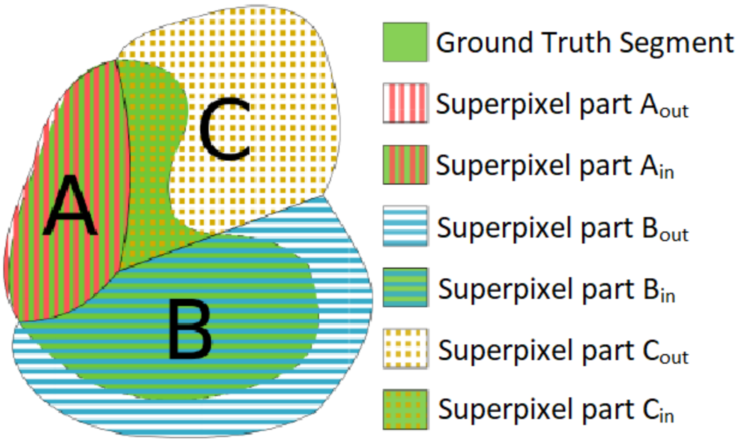

Figure 5.17: Three superpixels A, B, and C with their in and out parts flood over the ground truth segment border [2]

Undersegmentation error can be formulated as [43]:

$$Undersegmentation\, Error = \sum_{S \in GT} \frac{\sum_{P:P\cap S\neq\emptyset} |P_{\text{out}}|}{|S|} \tag{5.3}$$

where P represents the superpixels and S is the ground truth segment. Large superpixels, whose intersection area with ground truth segments is small, create high penalty values. Therefore, another formulation is proposed by Neubert and Protzel [2]:

$$Undersegmentation\, Error = \frac{1}{N}\left[\sum_{S \in GT}\left(\sum_{P:P\cap S\neq\emptyset} min(P_{\text{in}}, P_{\text{out}})\right)\right] \tag{5.4}$$

where N is the total number of pixels. This formulation is fairer than the other one because the erroneous segmentation should have caused by the minority of pixels in superpixel.

Consequently, a successful superpixel extraction algorithm should have high boundary recall value and low undersegmentation error.

5.4.2 Performance Metrics for Target Detection

The performance of the target detection algorithms is measured by using precision and recall metrics [44]. Precision can be defined as the rate of true positive samples among the retrieved samples. More specifically, the answer to the question, which is how many of the retrieved samples are positive, gives the precision. Recall is the fraction of retrieved true positive samples over the all positive samples in the ground truth. Namely, the answer to the question, which is how many of the positive samples are retrieved, gives the recall. These phenomena are illustrated in Figure 5.18.

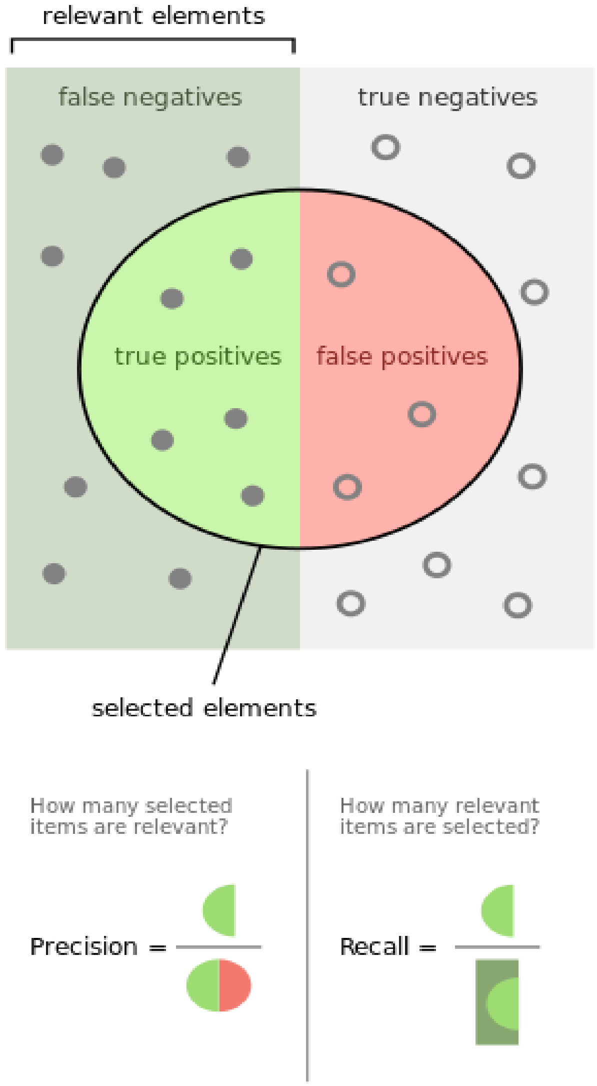

Figure 5.18: Illustration of precision and recall metrics [3]

Precision-recall curves provide the information for the detection capability of the algorithm. Precision value is shown in the y-axis of the precision-recall curve and its initial value is always 1. On the other hand, recall value is shown in the x-axis and its final value always reaches 1. In an ideal case, high precision with high recall value is expected. However, in real applications, the precision value decreases or remains constant while recall value is increasing.

The precision-recall curves can be obtained by using the score map of an algorithm and ground truth mask of the scene. After the score map is extracted, the number of thresholds should be determined to detect precision and recall values. The score values which are sampled by using the number of thresholds are the sampling points for the precision-recall curves. As a result, the ground truth mask and the scores are analyzed in terms of true positive, false positive, and precision values at these sampling points to create the precision-recall graph.

Depending on the applications, the user can decide which is more important among precision and recall values. If accurate detection of the little amount of the target material is important, then the high precision and the low recall values can be accepted. Conversely, if the detection of a large amount of the target material is a necessity then the high false alarm rates can be accepted due to the low precision and high recall values. The latter option can be applied to applications which can not tolerate any risk.

CHAPTER 6

EXPERIMENTAL RESULTS AND COMPARISONS

In this chapter, the experimental results for the superpixel extraction methods and the proposed target detection methods are given, compared and commented.

6.1 Superpixel Extraction Results

The results of the SLIC algorithm and boundary update-based superpixel extraction method change depending on the convexity weight parameters m and λ, and also the number of superpixels. To compare the results of these algorithms, different images and parameters are chosen. As the weight of the spectral distance is increasing, the regularity of the superpixels is decreasing. On the other hand, as the weight of the spatial distance is increasing, the adaptation of the superpixel boundaries to the object edges is decreasing. Therefore, the determination of the convexity weights is the most important step for the superpixel extraction methods. In this study, the spectral distance has more weight than the spatial distance due to importance of the segmentation of the similar spectral regions.

The superpixel extraction methods are applied to RGB images firstly. RGB images are chosen from Berkeley Segmentation Dataset and Benchmark due to the existence of ground truth for segmentation and boundaries [45]. After the superpixels are extracted, these superpixel boundaries are compared with the ground truth boundaries to calculate the boundary recall values for 1-pixel and 2-pixel distances. Also the undersegmentation error is calculated by using the superpixel boundaries. The best superpixel extraction algorithm should have high boundary recall and low undersegmentation error. The superpixel extraction results of the RGB images are compatible with the previous works [16].

After testing the superpixel algorithms on RGB images, Indian Pines and Salinas datasets are tested as the hyperspectral images. The ground truths which include segmented regions and boundaries of Indian Pines are given in Figure 6.1.

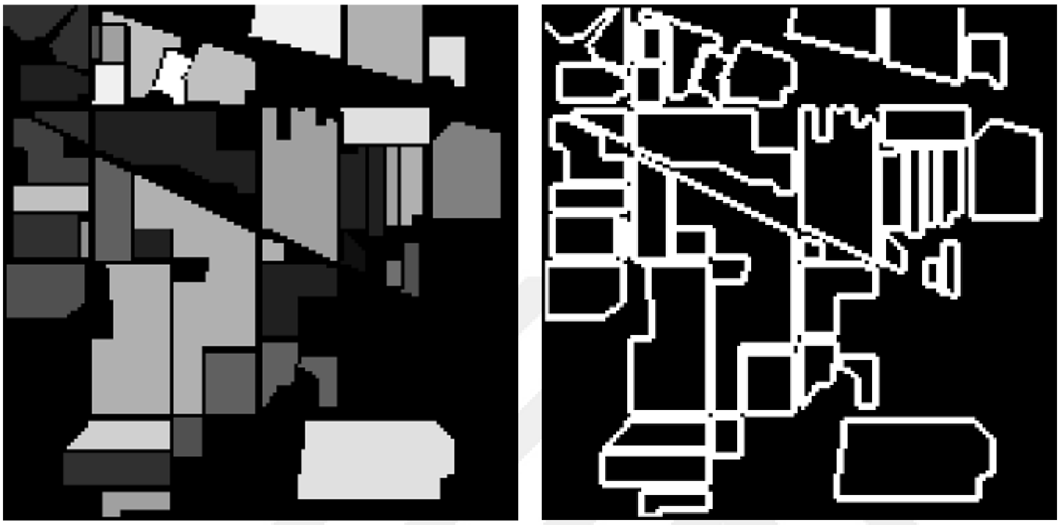

Figure 6.1: The ground truth segmentations and boundaries for the Indian Pines dataset

The SLIC algorithm and boundary update based superpixel extraction methods are applied to this dataset which has 145x145 pixel size for 225, 361, 441, 625 and 841 superpixels. The spectral distances of boundary update based superpixel extraction method are SAM, MSE, and SID which are explained in Section 4.1. The spectral distance weights of the boundary update based method are 0.99 and 0.995. These weights can be thought of as high but the result of the spatial distance is already high due to the calculation by using minimum square error. Therefore, the multiplications of weights and distances approach to the similar level by adjusting the weights. On the other hand, m parameters are chosen as 0.05 and 0.02 for the SLIC algorithm. These values are thought as convenient for this dataset. The superpixel extraction results of these algorithms for λ = 0.99, m = 0.05 and 441 superpixels are given in Figure 6.2 and 6.3. The corresponding superpixel boundaries are also given in Figure 6.4 and 6.5.

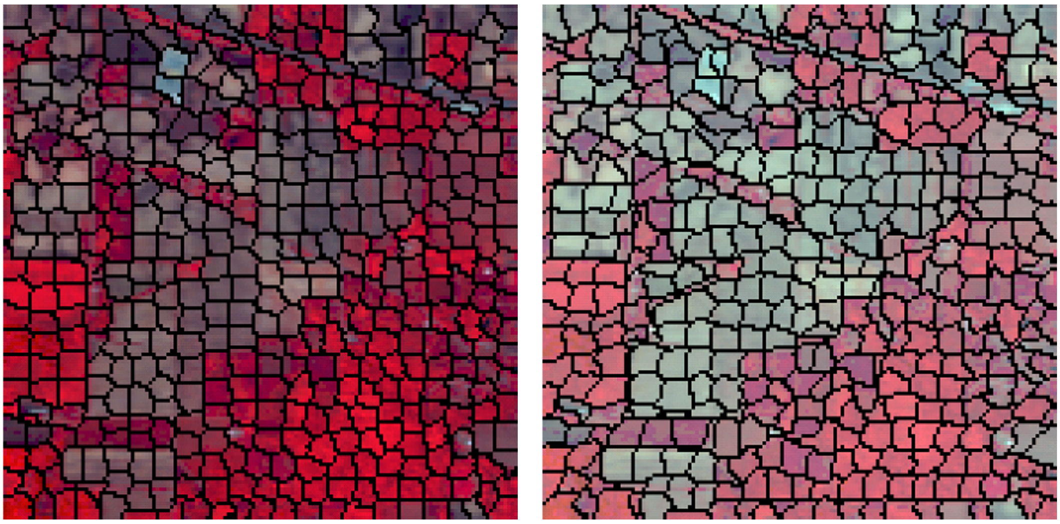

Figure 6.2: Superpixel results of a) the SLIC algorithm for $m = 0.05$ and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for $\lambda = 0.99$

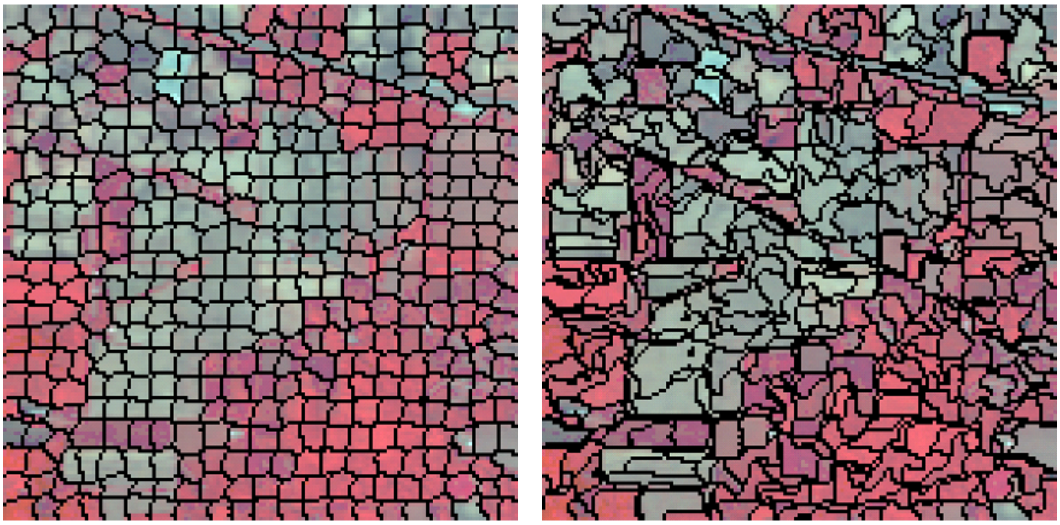

Figure 6.3: Superpixel results of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for $\lambda = 0.99$

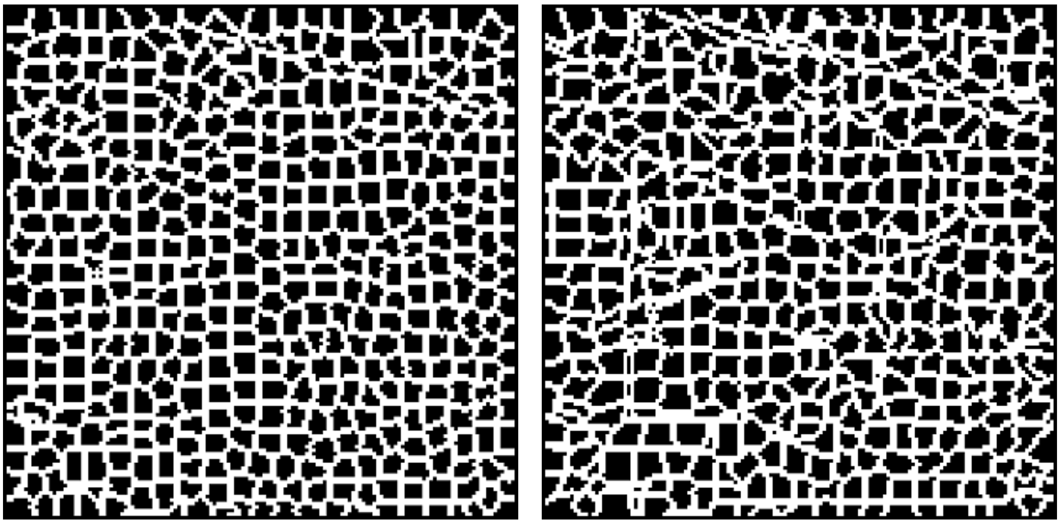

Figure 6.4: Superpixel boundaries of a) the SLIC algorithm for m = 0.05 and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for λ = 0.99

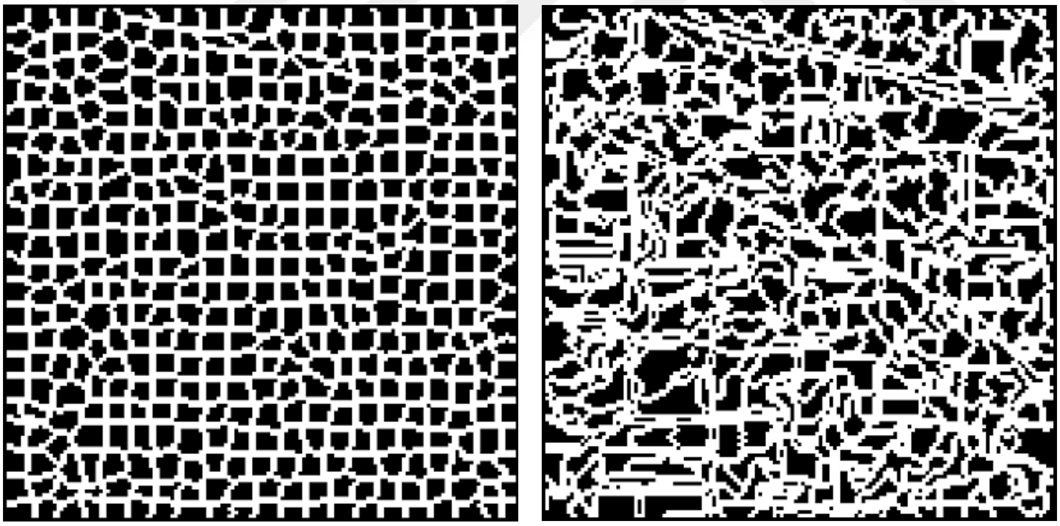

Figure 6.5: Superpixel boundaries of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for λ = 0.99

The performance metric results of the algorithms for 225, 361, 441, 625 and 841 superpixels are given in Figure 6.6 and 6.7.

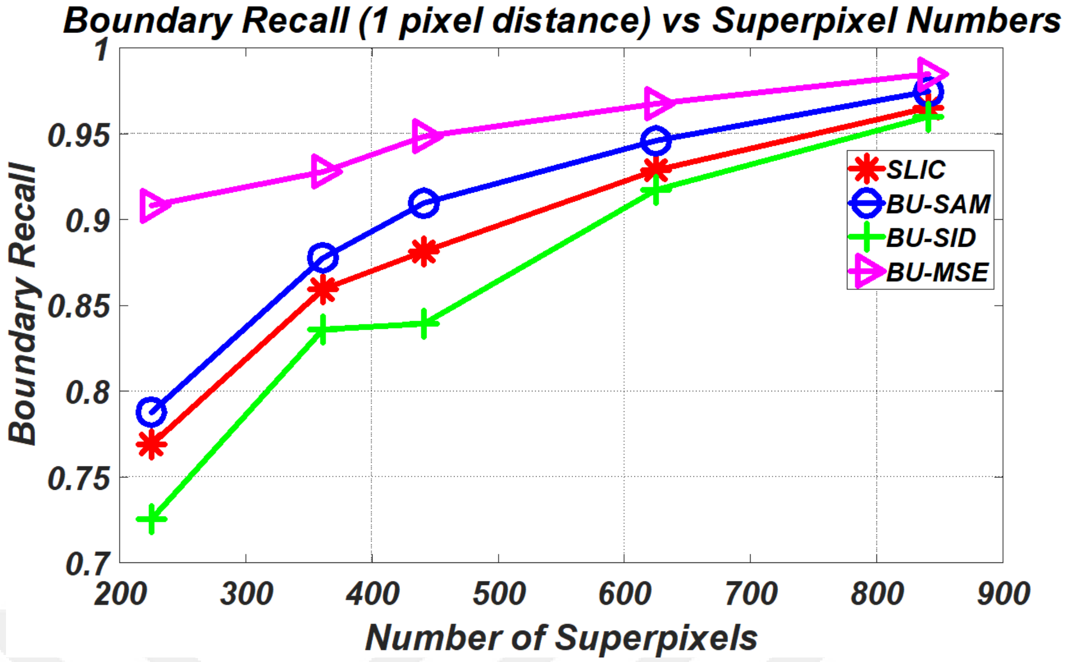

Figure 6.6: Boundary recall vs number of superpixels for λ = 0.99 and m = 0.05

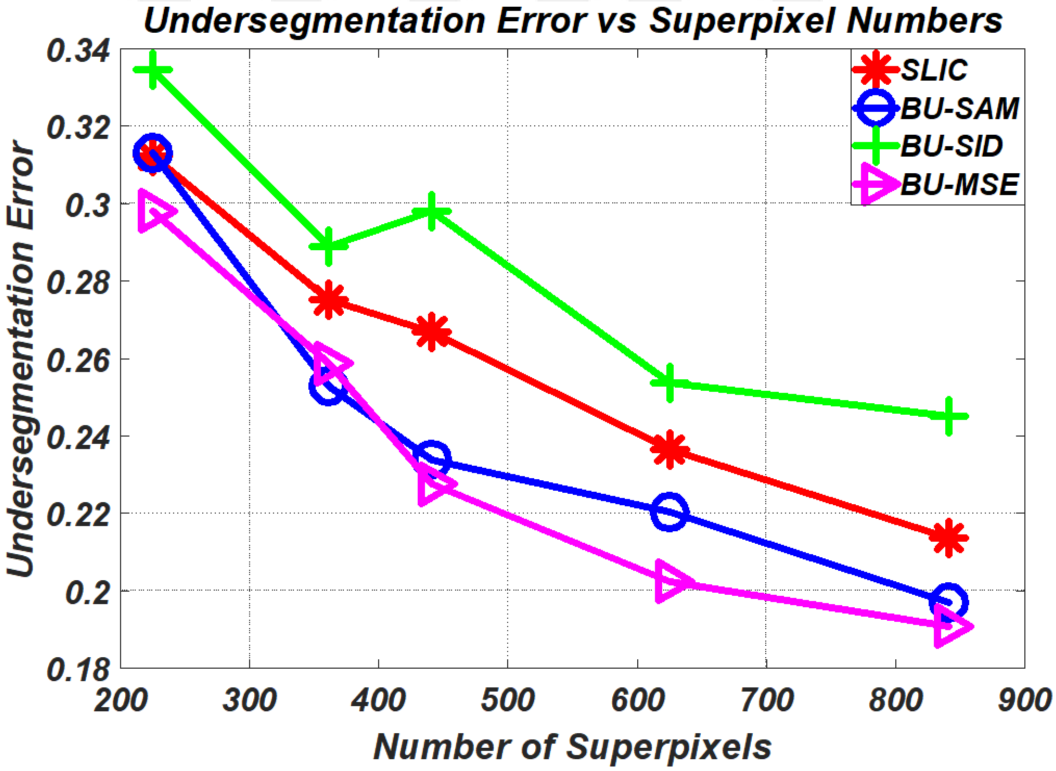

Figure 6.7: Undersegmentation error vs number of superpixels for λ = 0.99 and m = 0.05

The best algorithm for these parameters is the boundary update based superpixel extraction method using MSE as the spectral distance metric. Although both of the SLIC algorithm and the boundary update based superpixel extraction method using MSE as the spectral distance metric are thought as the similar algorithms, their approaches to the creating of superpixels are totally different. While the SLIC algorithm is computing the distances for all pixels in $2Sx2S$ area, the boundary update based superpixel extraction method computes the distances between boundary pixels and the superpixel centers. Therefore, the performance results of these algorithms are distinguished. Also, the boundary update based method using SAM is better than the SLIC algorithm. Using different spectral distance metrics and superpixel numbers affect the results for the same weight parameter. As a clear observation, increasing the number of superpixels gives the better results due to the better fitting of the superpixel boundaries to the edges.

The superpixel extraction results of these algorithms for λ = 0.995, m = 0.02 and 441 superpixels are given in Figure 6.8 and 6.9. The corresponding superpixel boundaries are also given in Figure 6.10 and 6.11.

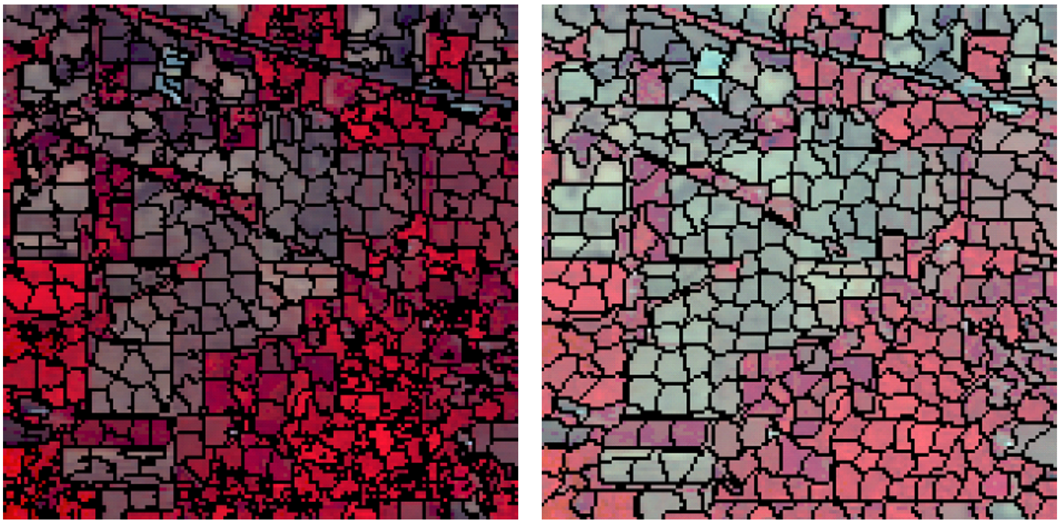

Figure 6.8: Superpixel results of a) the SLIC algorithm for m = 0.02 and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for λ = 0.995

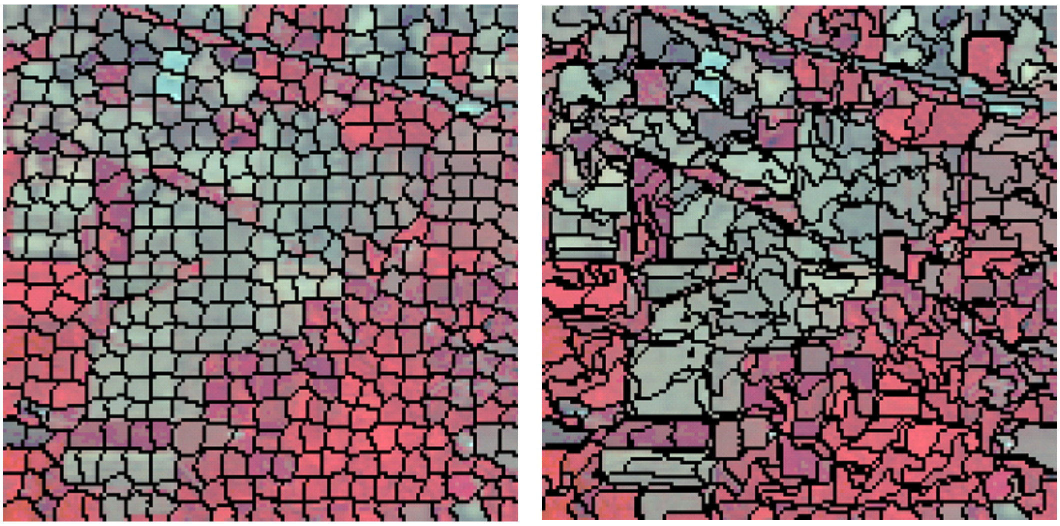

Figure 6.9: Superpixel results of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for $\lambda = 0.995$

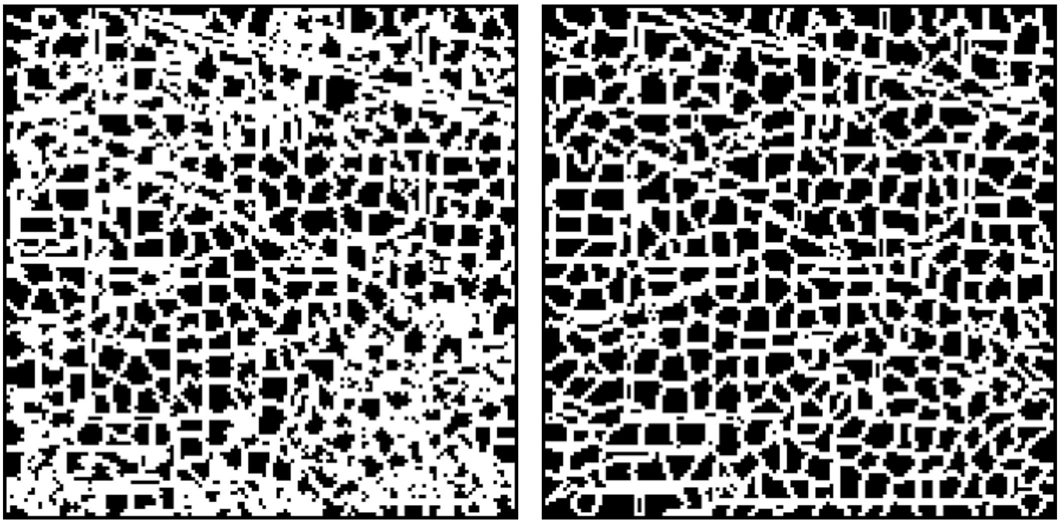

Figure 6.10: Superpixel boundaries of a) the SLIC algorithm for m = 0.02 and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for $\lambda = 0.995$

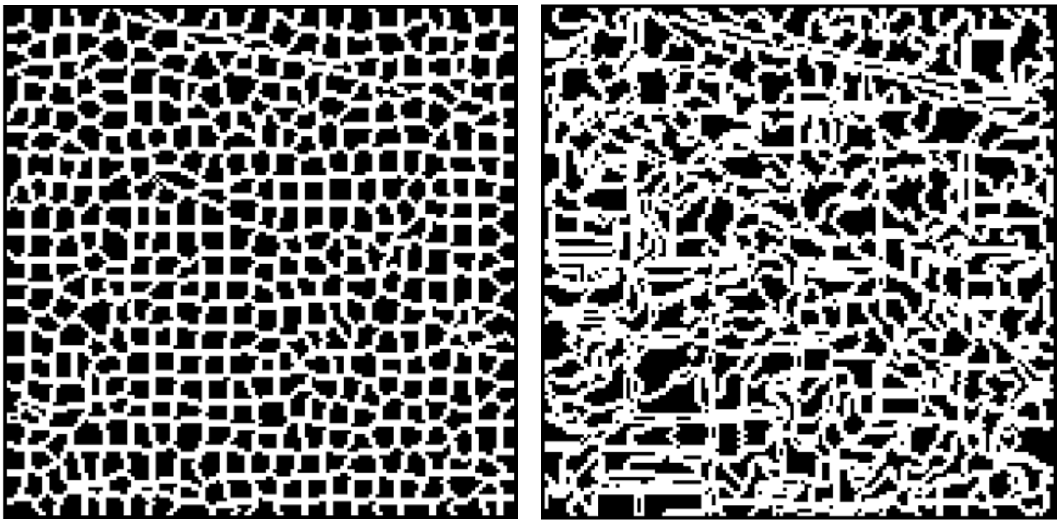

Figure 6.11: Superpixel boundaries of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for $\lambda = 0.995$

The performance metric results of the algorithms for $\lambda = 0.995$ and $m = 0.02$ are given in Figure 6.12 and 6.13.

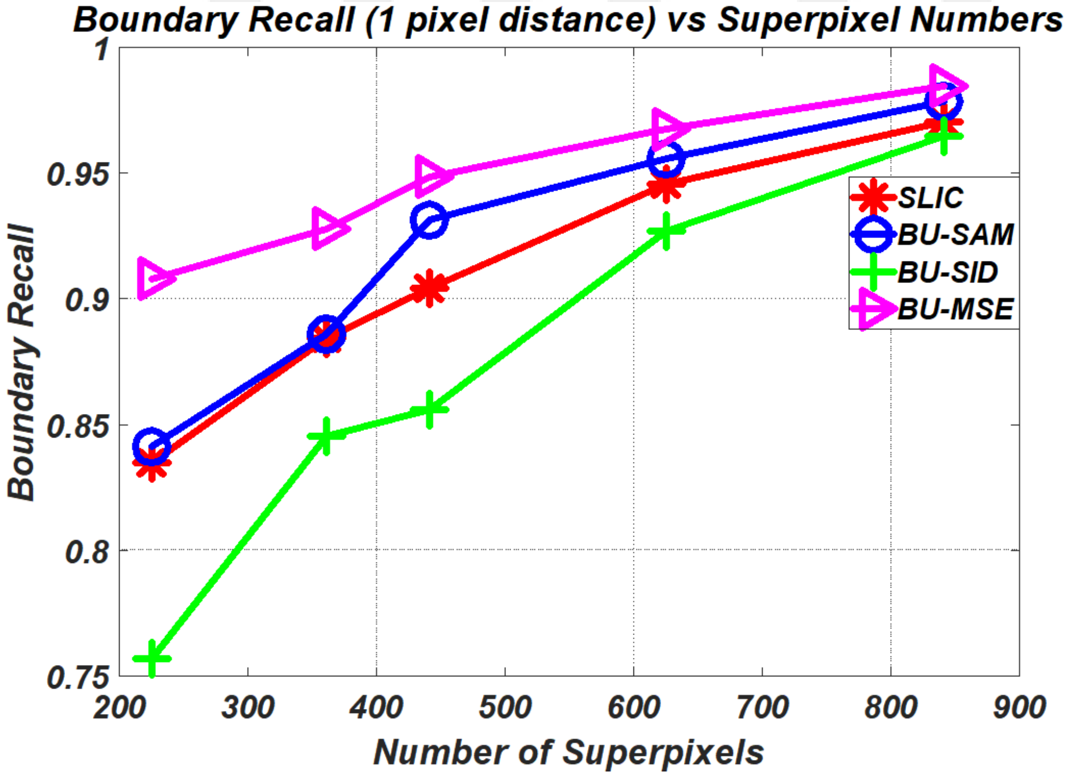

Figure 6.12: Boundary recall vs number of superpixels for $\lambda = 0.995$ and $m = 0.02$

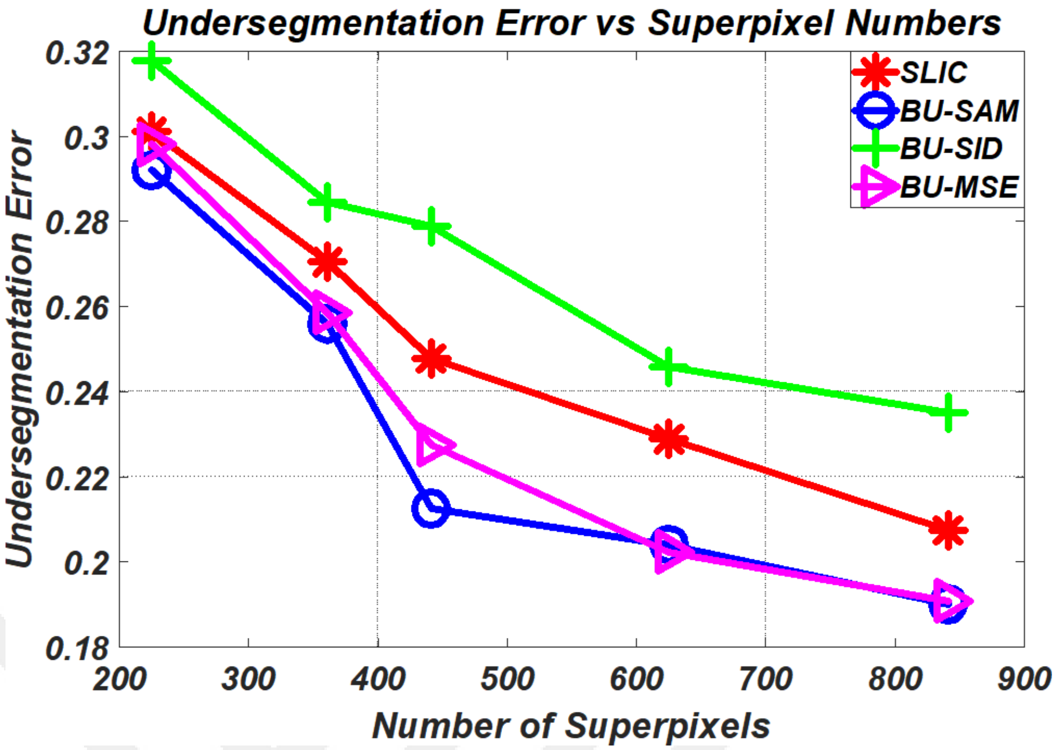

Figure 6.13: Undersegmentation error vs number of superpixels for λ = 0.995 and m = 0.02

The results are similar to the previous result. However, the performance of the boundary update based superpixel extraction method using SAM as the spectral distance increases in terms of undersegmentation error. The importance of the weight parameter can be understood by comparing these two results. The slight difference in the weight parameter can cause the dissimilarity in the performances of the algorithms. Therefore, these results are highly correlated with the selection of the weight parameters. The best two algorithms are boundary update based superpixel extraction methods for these weight parameters. Nevertheless, the boundary update based superpixel extraction method using SID as the spectral distance cannot be described as the worst algorithm. The chosen weight parameter might not be the optimal parameter for the SID.

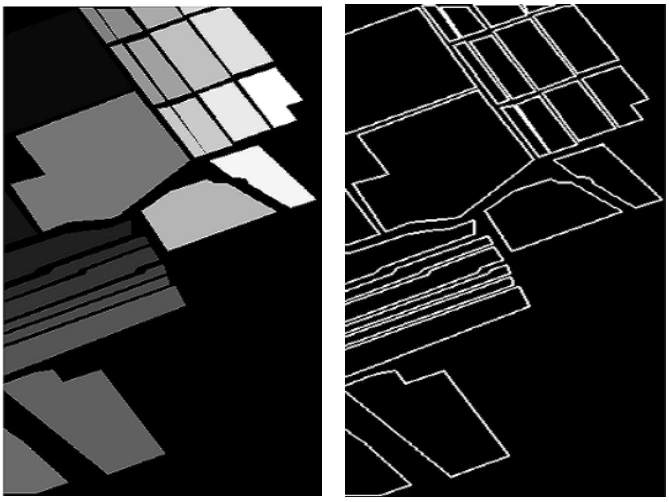

Figure 6.14: The ground truth segmentations and boundaries for the Salinas dataset

The ground truths in terms of segmented regions and boundaries of Salinas dataset are given in Figure 6.14. The Salinas dataset has 512x217 pixel size with 204 spectral bands. Superpixel extraction algorithms are applied to this dataset for 1144, 1792, 2294, 3182, and 4532 superpixels. The superpixel extraction results of these algorithms for $\lambda = 0.98$, $m = 0.1$ and 2294 superpixels are given in Figure 6.15 and 6.16.

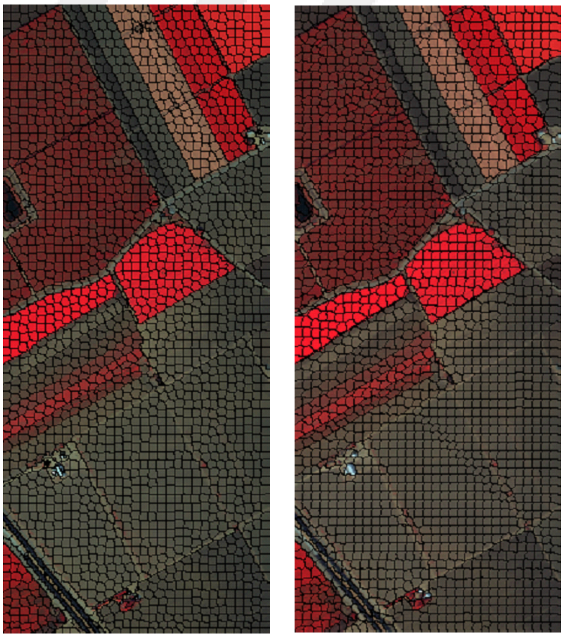

Figure 6.15: Superpixel results of a) the SLIC algorithm for $m = 0.1$ and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for $\lambda = 0.98$

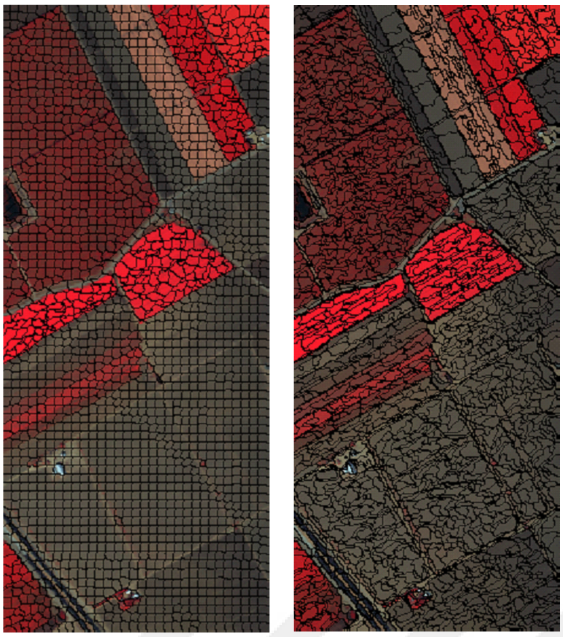

Figure 6.16: Superpixel results of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for $\lambda = 0.98$

The corresponding superpixel boundaries are given in Figure 6.23 and 6.24.

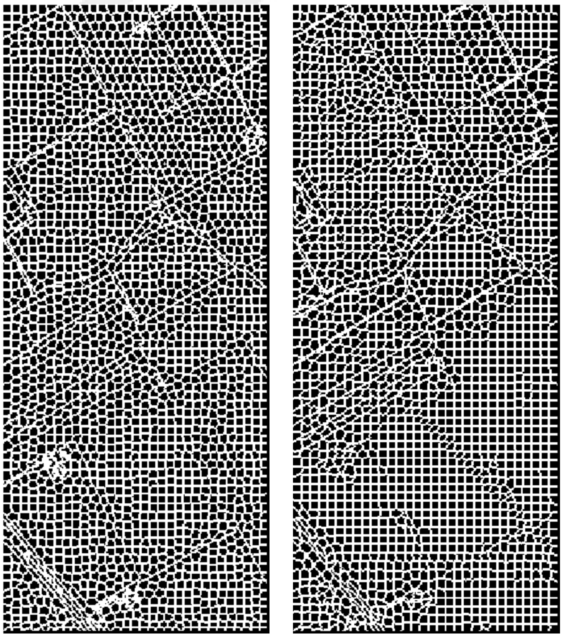

Figure 6.17: Superpixel boundaries of a) the SLIC algorithm for $m = 0.1$ and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for $\lambda = 0.98$

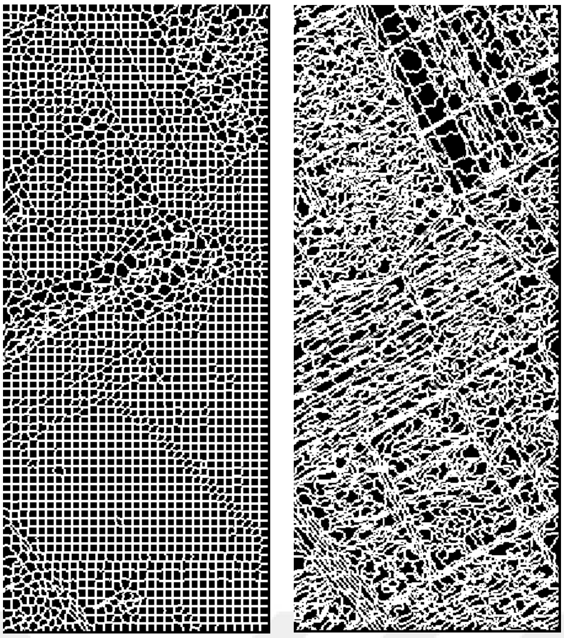

Figure 6.18: Superpixel boundaries of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for $\lambda = 0.98$

The performance metric results of the algorithms for 1144, 1792, 2294, 3182 and 4532 superpixels are given in Figure 6.19 and 6.20.

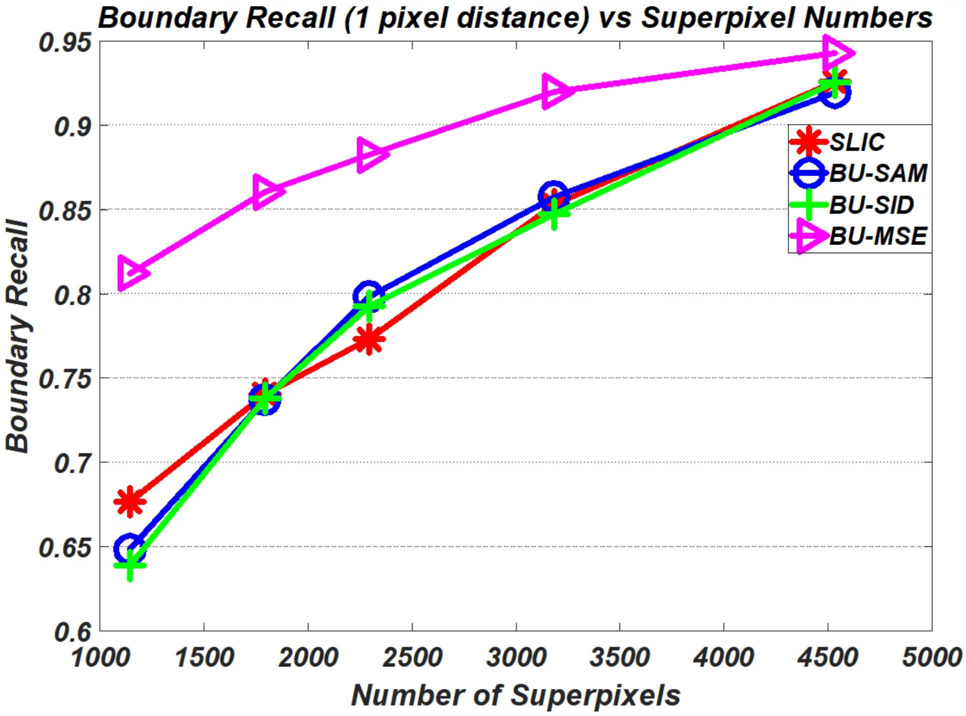

Figure 6.19: Boundary recall vs number of superpixels for $\lambda = 0.98$ and $m = 0.1$

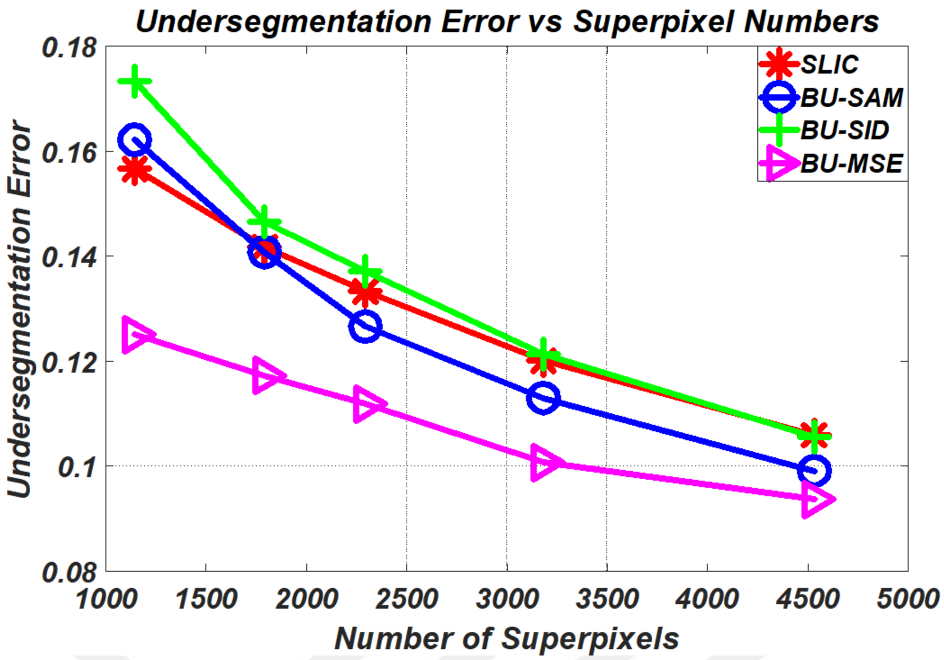

Figure 6.20: Undersegmentation error vs number of superpixels for λ = 0.98 and m = 0.1

The boundary update based method using MSE as the spectral distance is the best option according to the performance metric results. The results are convenient with the previous results. Comparing the SLIC algorithm with the boundary update based superpixel extraction method using MSE as the spectral distance is fairer than the other comparisons due to the application of the same spectral distance. As it can be inferred from the results, superpixel extraction by computing the distance metrics between the boundary pixels and the superpixel centers provides better results than the SLIC algorithm. Four comparisons at most between pixels are computing in the boundary update based superpixel extraction method. Therefore, the boundary pixel assignment to the nearest superpixel centers provides less computational complexity and more accurate segmentation.

The number of superpixels is directly proportional to the boundary recall and inversely proportional to the undersegmentation error. However, as the number of superpixels increases, the computational complexity of the algorithms increases as well.

The superpixel extraction results of these algorithms for λ = 0.99, m = 0.05 and 2294 superpixels are given in Figure 6.21 and 6.22.

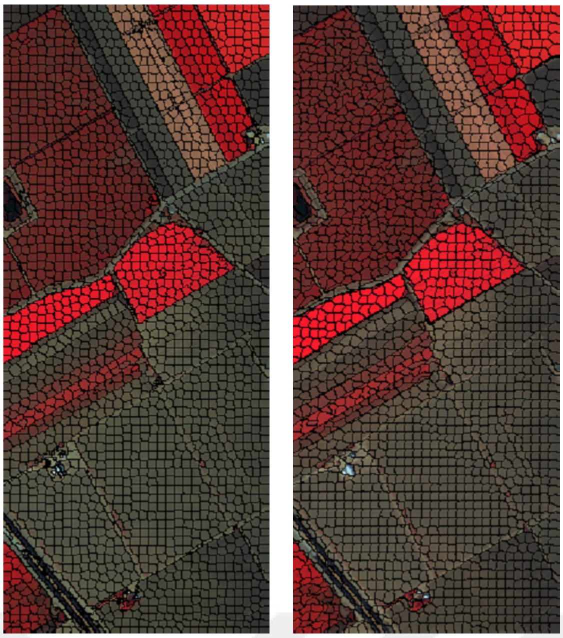

Figure 6.21: Superpixel results of a) the SLIC algorithm for $m = 0.05$ and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for $\lambda = 0.99$

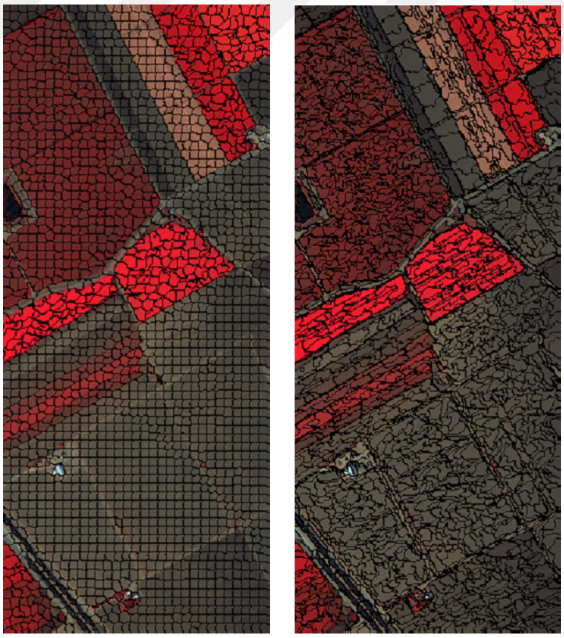

Figure 6.22: Superpixel results of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for $\lambda = 0.99$

The corresponding superpixel boundaries are given in Figure 6.23 and 6.24.

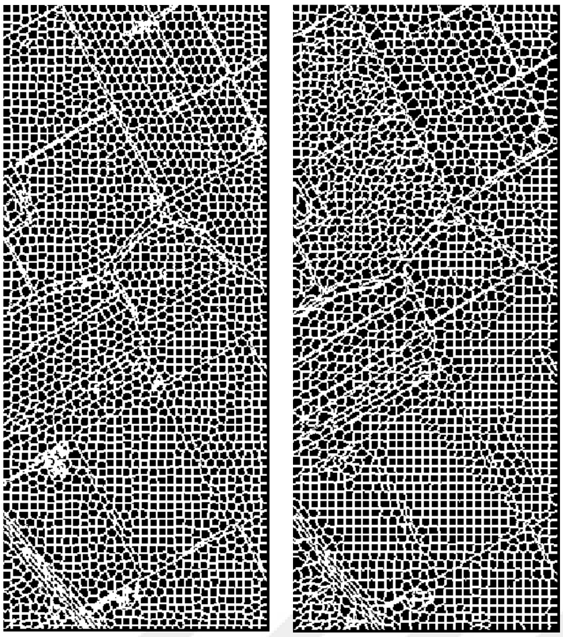

Figure 6.23: Superpixel boundaries of a) the SLIC algorithm for m = 0.05 and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for λ = 0.99

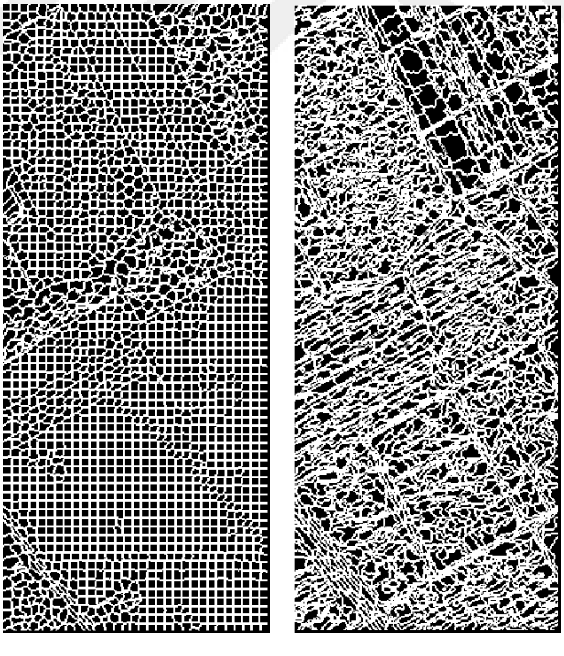

Figure 6.24: Superpixel boundaries of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for λ = 0.99

The performance metric results of the algorithms for 1144, 1792, 2294, 3182 and

4532 superpixels are given in Figure 6.25 and 6.26.

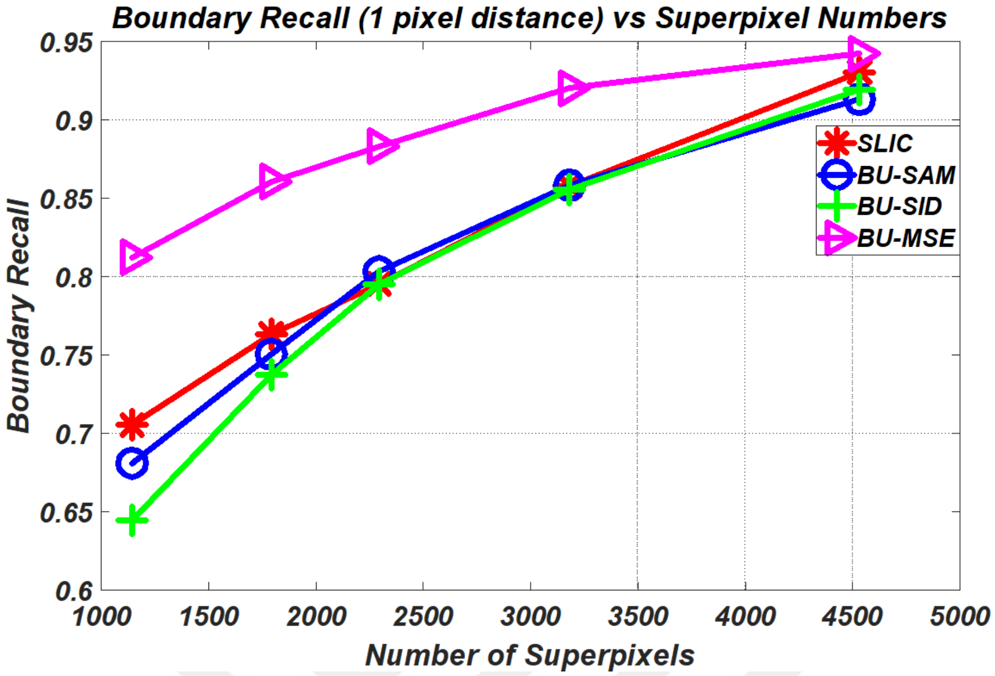

Figure 6.25: Boundary recall vs number of superpixels for λ = 0.99 and m = 0.05

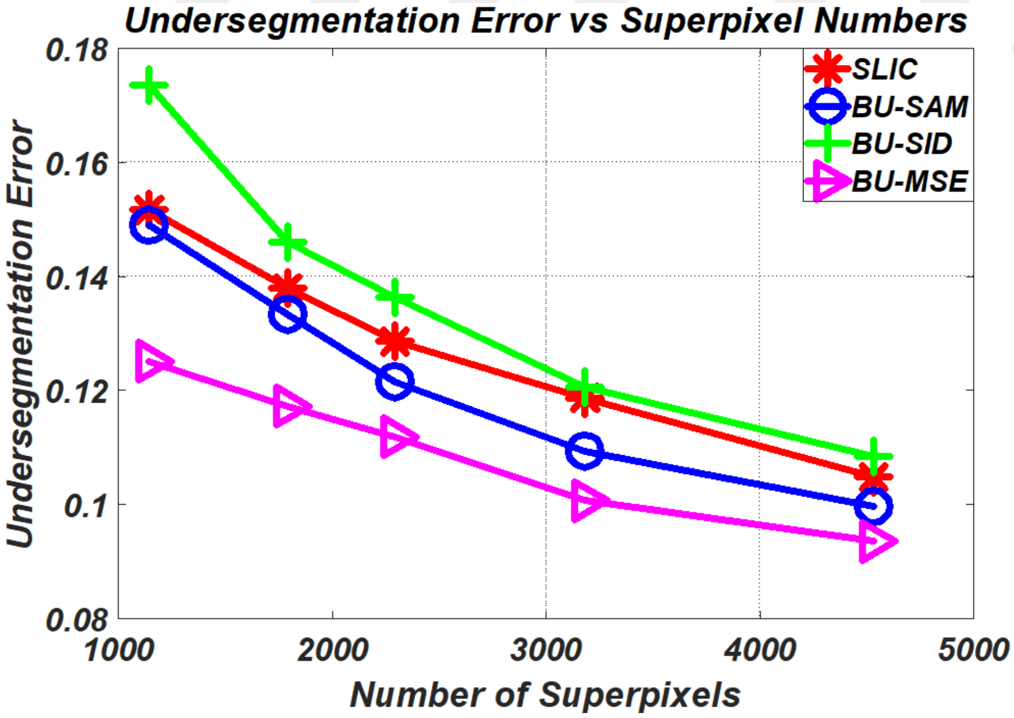

Figure 6.26: Undersegmentation error vs number of superpixels for λ = 0.99 and m = 0.05

The boundary update based superpixel extraction method using MSE as the spectral distance is superior to the other algorithms for these weight parameters. All of the algorithms have similar results with the previous one.

As the last test of superpixel extraction algorithms, the dataset shown in Figure 5.15 is used. The ground truth segmentation, ground truth boundaries, superpixel extraction results and superpixel boundaries are given below. This dataset has 512x640 pixel size. As an input of superpixel extraction algorithms, the hyperspectral image has 71 bands within the interval of 1000-1700 nm. Superpixel extraction algorithms are applied to this dataset for 1280, 1702, 2322, 3328 and 5120 superpixels. The superpixel extraction results of these algorithms for λ = 0.99, m = 0.2 and 2322 superpixels are given in Figure 6.28 and 6.29.

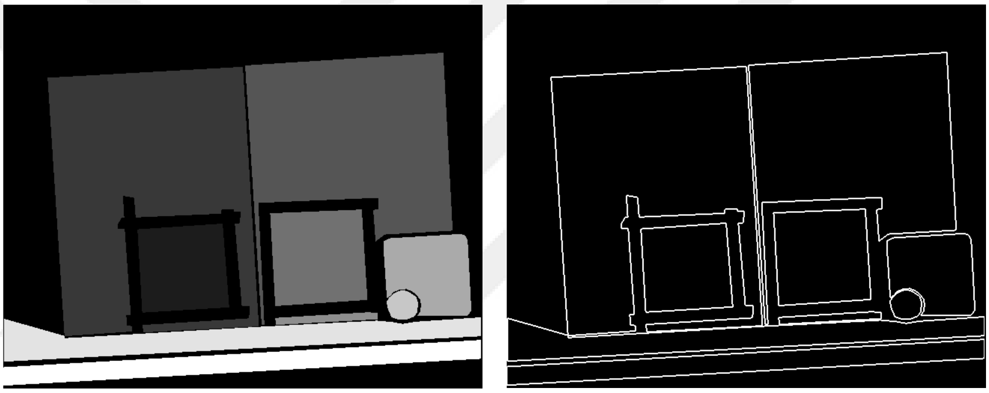

Figure 6.27: The ground truth segmentations and boundaries for the prepared dataset

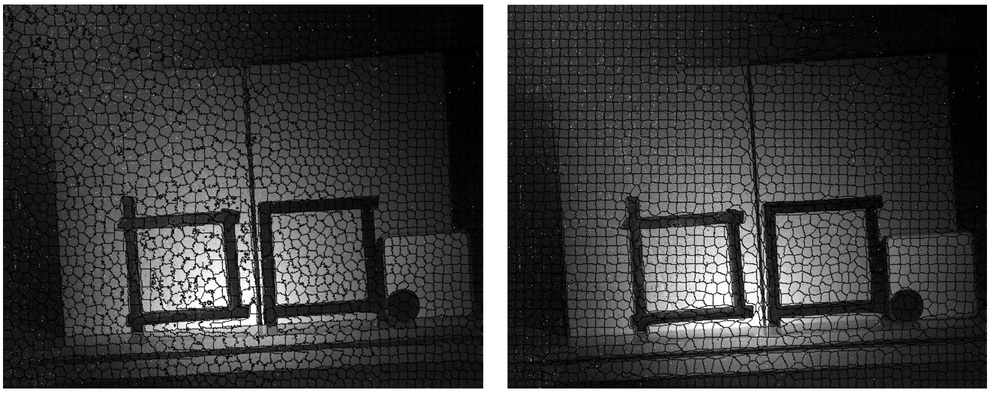

Figure 6.28: Superpixel results of a) the SLIC algorithm for m = 0.2 and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for λ = 0.99

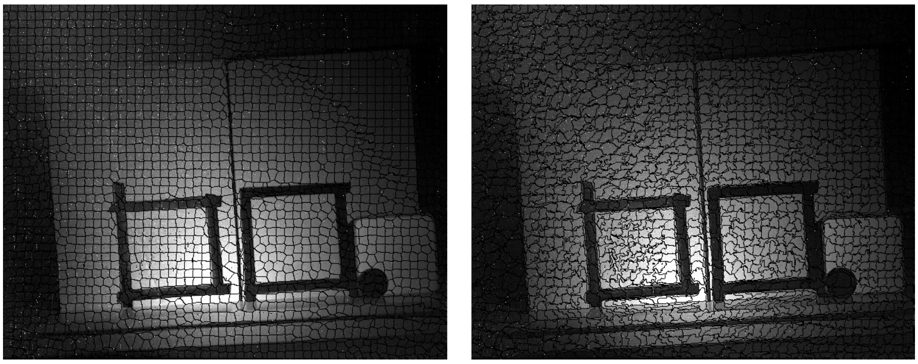

Figure 6.29: Superpixel results of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for λ = 0.99

The corresponding superpixel boundaries are given in Figure 6.30 and 6.31.

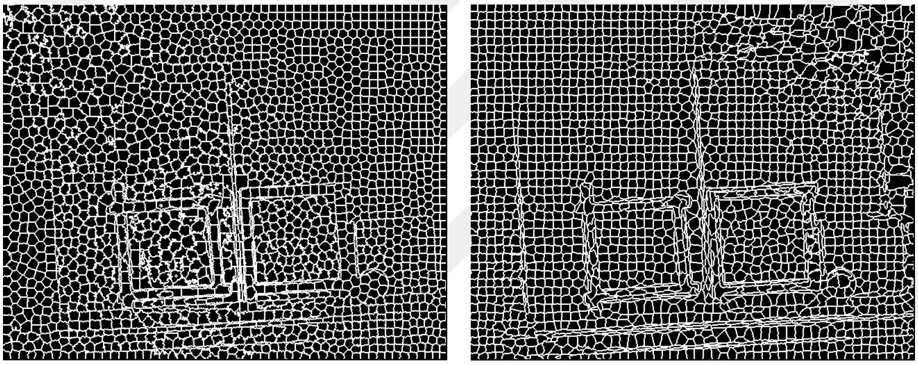

Figure 6.30: Superpixel boundaries of a) the SLIC algorithm for m = 0.2 and b) boundary update based superpixel extraction method that uses SAM as the spectral distance for λ = 0.99

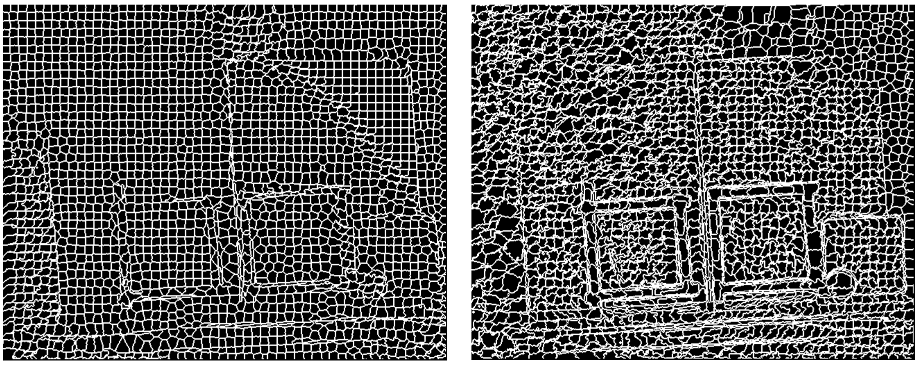

Figure 6.31: Superpixel boundaries of boundary update based superpixel extraction method that uses a) SID and b) MSE as the spectral distances for λ = 0.99

The performance metric results of the algorithms for 1280, 1702, 2322, 3328 and 5120 superpixels are given in Figure 6.32 and 6.33.

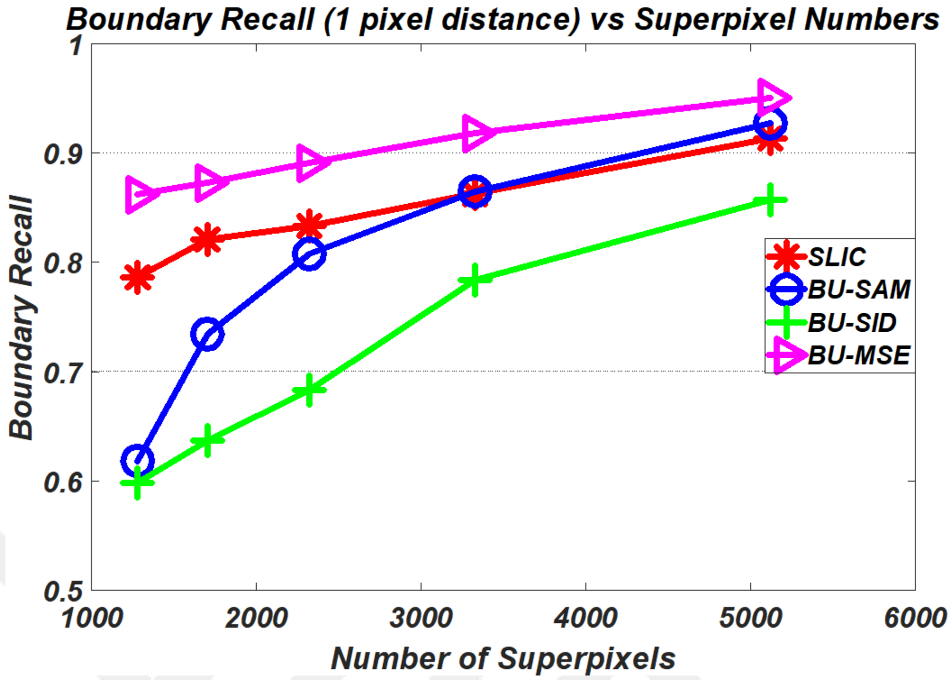

Figure 6.32: Boundary recall vs number of superpixels for λ = 0.99 and m = 0.2

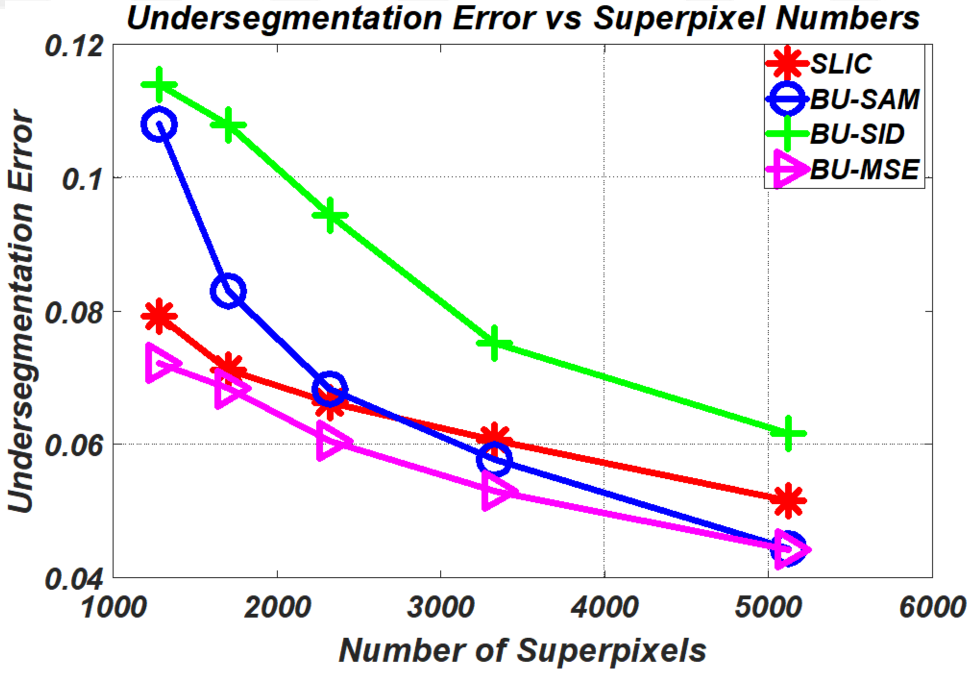

Figure 6.33: Undersegmentation error vs number of superpixels for λ = 0.99 and m = 0.2

These results are compatible with the previous tests. Therefore, it can be inferred from the experiments that the boundary update based superpixel extraction method can be applied to datasets instead of the SLIC algorithm compared to the results of the algorithms in terms of the boundary recall and undersegmentation error metrics.

6.2 Target Detection Results

In this section, the results of the signature-based and superpixel-based target detection algorithms are given and compared in terms of precision and recall values. On the other hand, the effects of the hybrid detectors and their superpixel adapted versions are analyzed and compared to traditional approaches. Firstly, RIT dataset is tested for the performance of target detection algorithms[39,46]. This dataset includes 280x800 pixels with 126 spectral bands. Different target materials were placed on the ground and hyperspectral image was captured by an airplane. These target materials are fabric panels and vehicles. Fabric panel shown in Figure 6.34 is selected as the target in the experiment.

Figure 6.34: Fabric panel-target material of RIT dataset

The spectral range of the camera used for this dataset is 450-2500 nm. Positions and the spectral signatures of the target materials are available in the dataset and shown below. Three regions of interests were given in the dataset: full-pixel, sub-pixel, and guard. This fabric panel covers one full-pixel and eight sub-pixels which can be regarded as a very small area in the image consisting of 224000 pixels. Therefore, the detection of this target is very difficult. Moreover, the average superpixel area must be 3x3 due to the coverage area of the fabric panel. For this reason, the number of superpixels is chosen as 25000 for this dataset.

Figure 6.35: Full-pixel and sub-pixel positions of the fabric panel

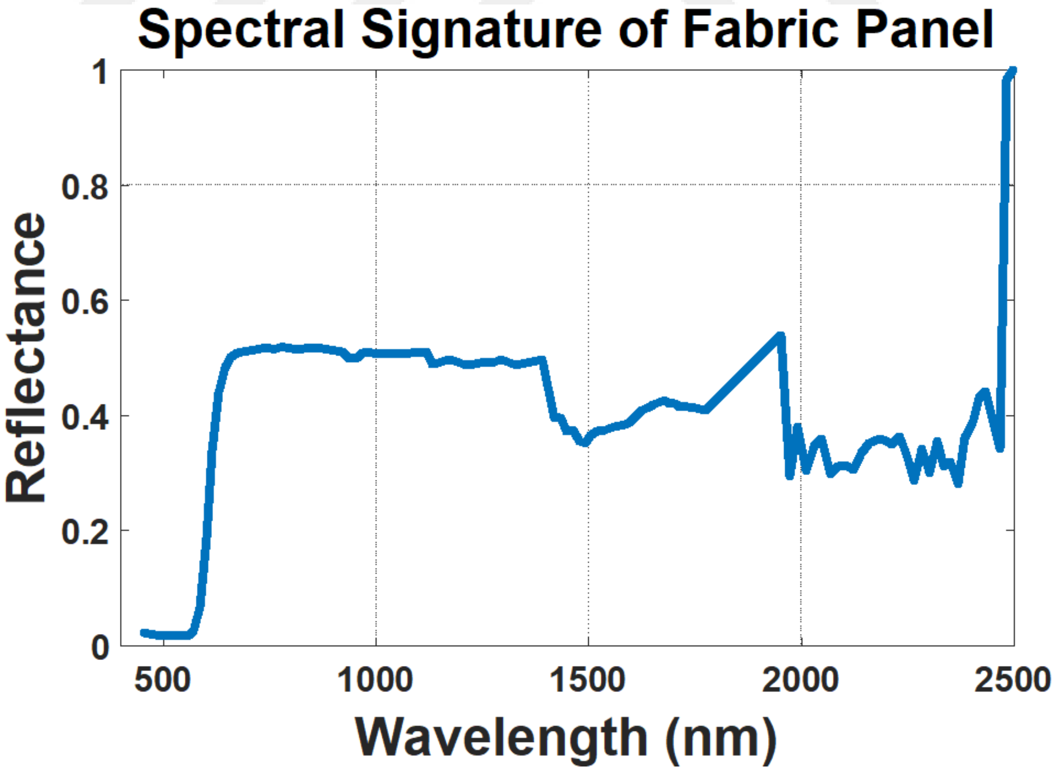

Figure 6.36: Spectrum of the fabric panel

This spectral signature is given as an input to the traditional signature-based target detection algorithms such as SAM, ACE and SMF. These algorithms apply their statistical analyses to individual pixels. As a result, each pixel has a score which defines the similarity of the pixel spectrum with the target spectrum. A threshold value is applied for discriminating the most similar pixels from other pixels. All of the target detection algorithms except the SAM algorithm result in high scores for similar pixels. As mentioned before, the SAM algorithm shows the angle between pixels and the target spectrum. Therefore, the less angle (or the score of the SAM algorithm) means more similarity. Ground truth mask for the target material is shown in Figure 6.37. Also, the results of the best algorithms for this dataset are given below as score images and their thresholded versions.

Figure 6.37: Ground truth mask of fabric panel in RIT dataset

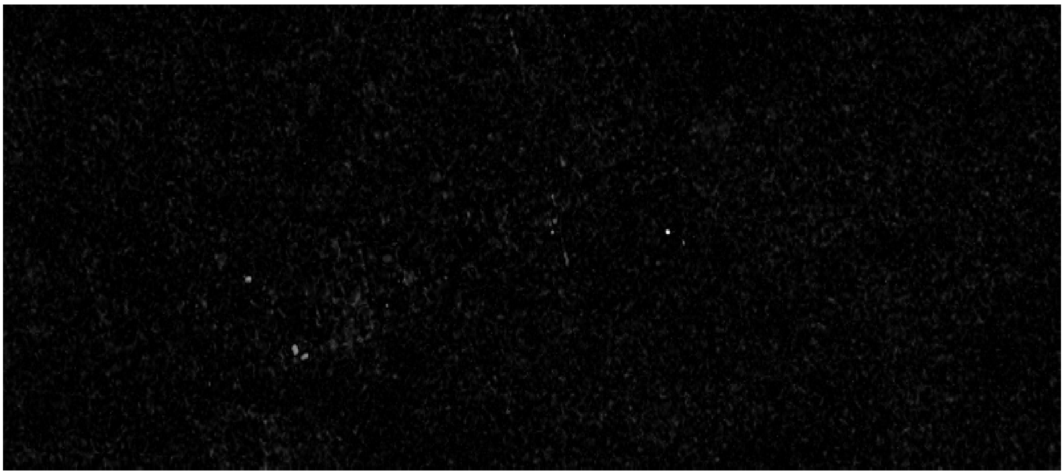

Figure 6.38: ACE score image for RIT dataset

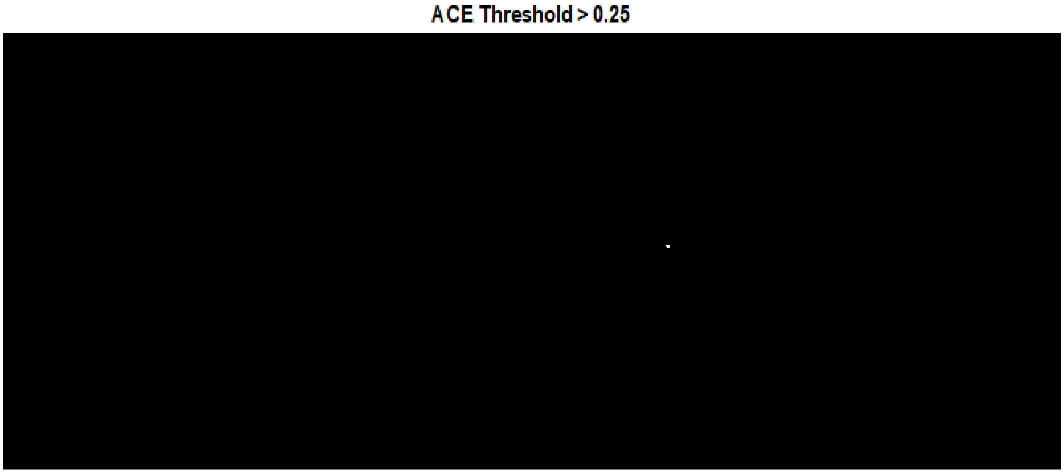

Figure 6.39: ACE score image with 0.25 threshold

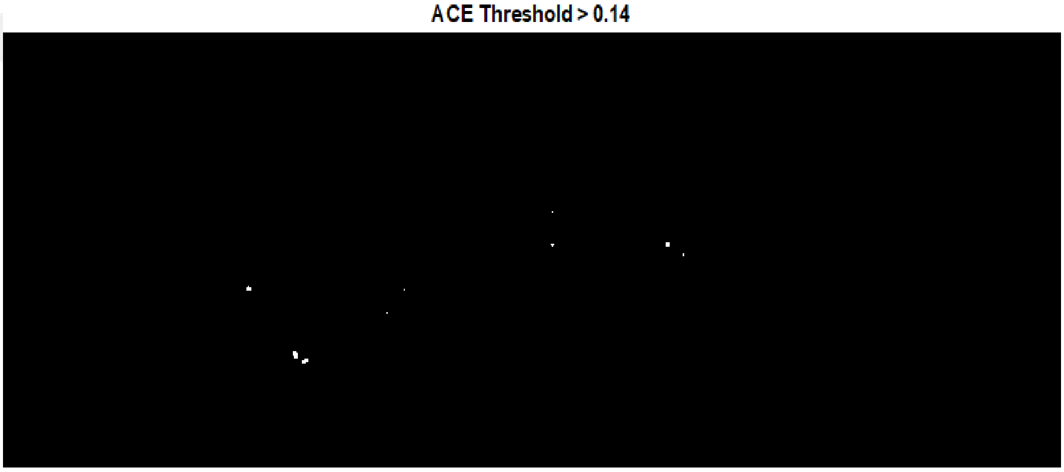

Figure 6.40: ACE score image with 0.14 threshold

The results of the ACE algorithm are quite satisfactory. The full-pixel target has the highest score which is 0.39. There are 5 true positive pixels and no false positive pixels in the result when the threshold value is chosen as 0.25. At this threshold value, the precision value is 1 and the recall value is 0.56 (5 true positive pixels out of 9 pixels). On the other hand, if the threshold value decreases to 0.14, all of the target pixels and some false positive pixels are obtained in the result. At this threshold value, the precision value decreases to 0.16 (9 true positive pixels out of 55 pixels) and the recall value increases to 1. Choosing a threshold value of 0.25 is the best option for this algorithm because of the higher precision value with satisfactory recall value. Also it should be noted that using precision-recall curves are pointless for this dataset, because of the few numbers of pixels consisting of the target material.

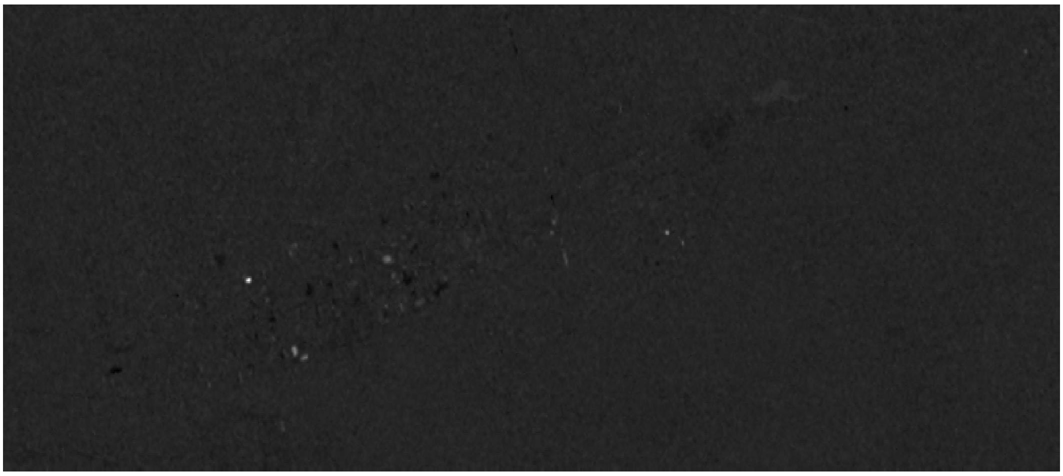

Figure 6.41: SMF score image for RIT dataset

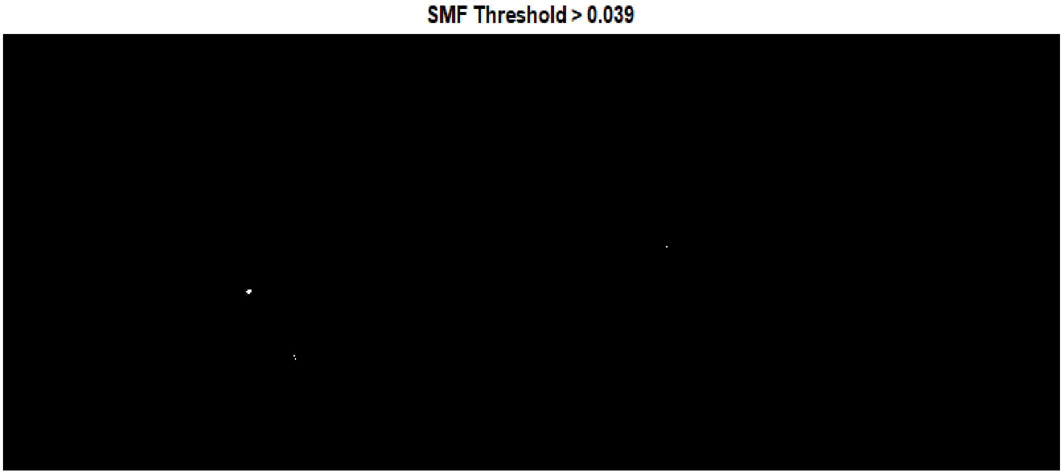

Figure 6.42: SMF score image with 0.039 threshold

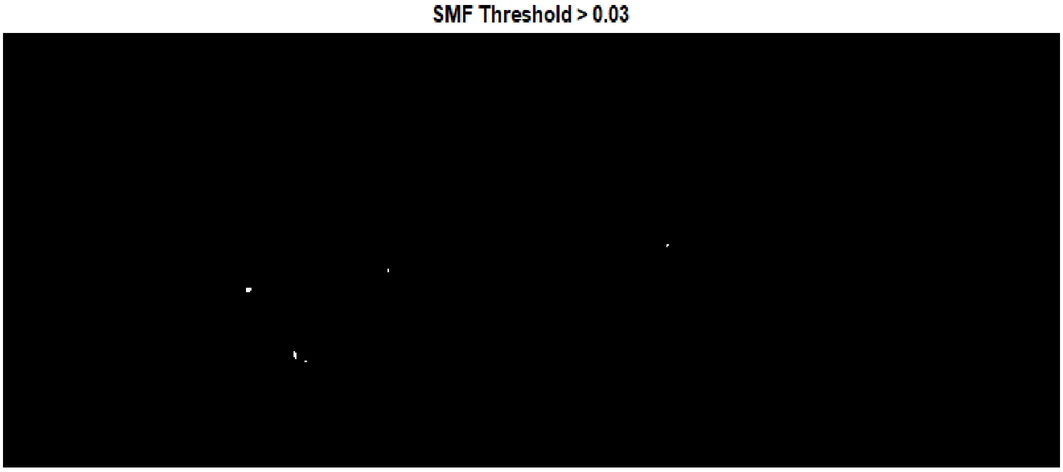

Figure 6.43: SMF score image with 0.03 threshold

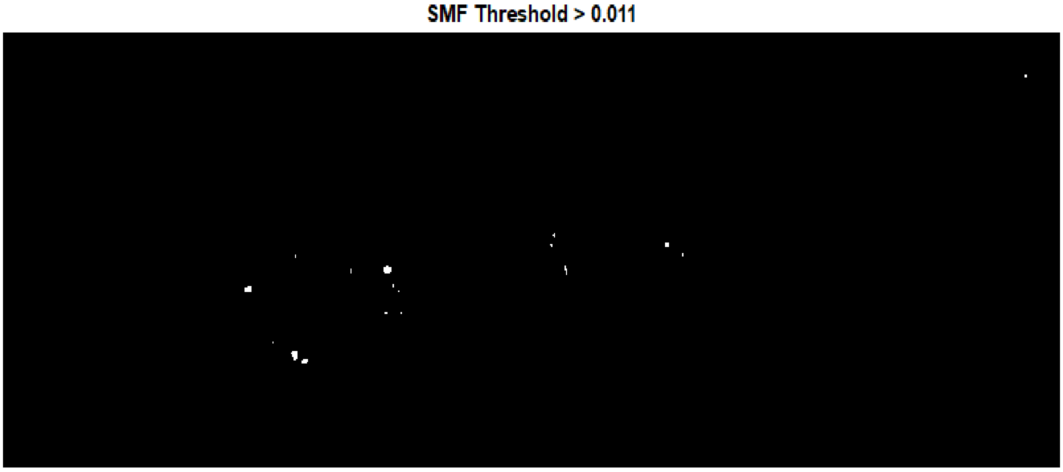

Figure 6.44: SMF score image with 0.011 threshold

The results of the SMF algorithm are not as good as the ACE algorithm. There is 1 true positive pixel which is full-pixel target when the threshold value is chosen as 0.039. Also, there are some false positive pixels in the result. At this threshold value, the precision value is 0.083 (1 true positive pixel out of 12) and the recall value is 0.11 (1 true positive pixel out of 9 pixels). The second threshold value is chosen as 0.03. The precision value increases to 0.11 (3 true positive pixels out of 26) and the recall value increases to 0.33 (3 true positive pixels out of 9 pixels) at this threshold level. The last threshold value is chosen for reaching to the highest recall value which is 1. However, the precision value decreases to 0.076 (9 true positive pixels out of 118 pixels) at this level.

The best algorithms of the traditional target detection algorithms are ACE and SMF for this dataset. ACE is superior to SMF algorithm in terms of precision and recall values. The background modeling of the ACE algorithm is more realistic than the SMF algorithm because it is done by taking different scaled covariance matrices for background and target.

The working principle of hybrid detectors is different than traditional target detection algorithms. Hybrid detectors can decrease the number of false positive pixels by applying both unmixing algorithms and target detection algorithms. However, in order to apply hybrid detectors, background signatures are also required. 9 pixels are chosen from the image to create background signature matrix. The results of the best hybrid detector for this dataset are given below as score images and thresholded versions.

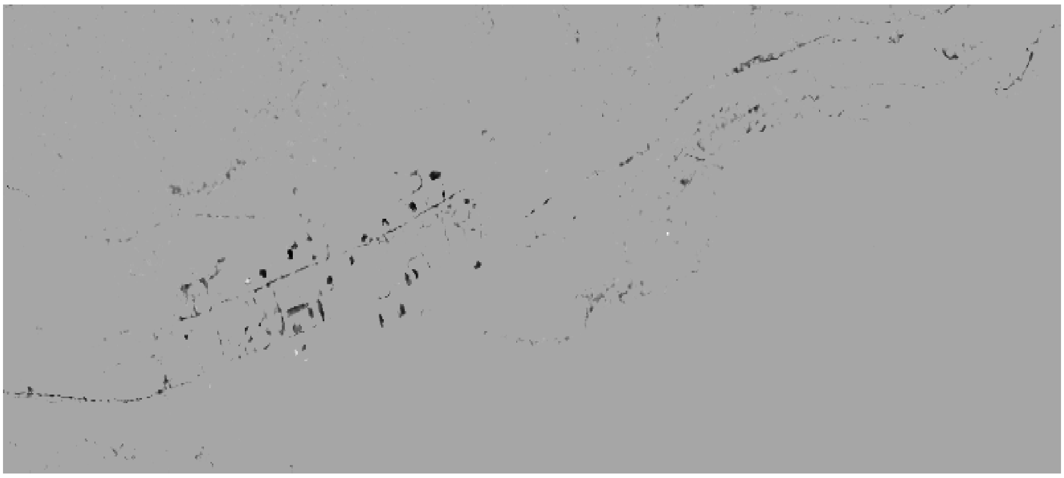

Figure 6.45: HSD score image for RIT dataset

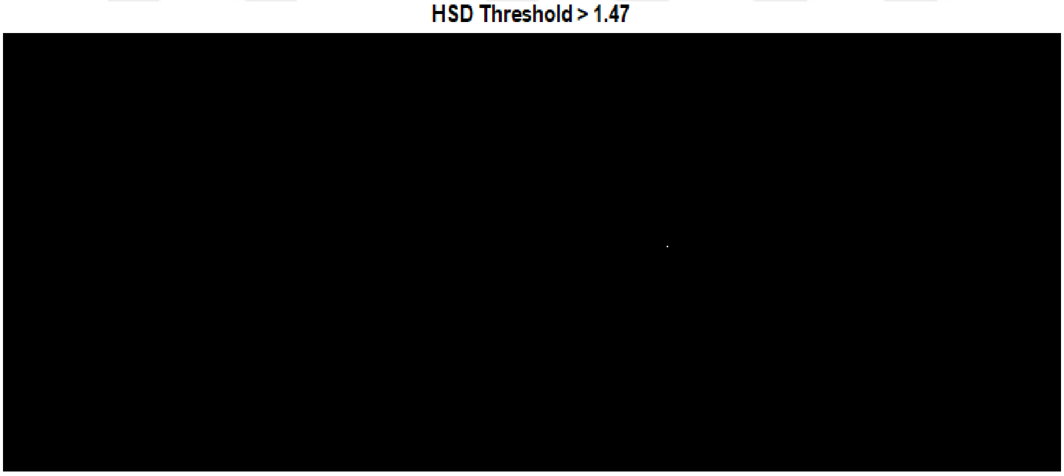

Figure 6.46: HSD score image with 1.47 threshold

As it can be inferred from the results of the HSD algorithm, the background is highly suppressed. Although there are some false positive pixels in the results, their number is less than in the previous results. The full-pixel target has the highest score and it can be seen in Figure 6.46. The precision value is 0.21 (3 true positive pixels out of 14 pixels) and the recall value is 0.33 (3 true positive pixels out of 9 pixels) for the threshold value of 1.169. The third threshold value is 1.09. The precision is 0.167 (4 true positive pixels out of 24 pixels) and the recall is 0.44 (4 true positive pixels out of 9 pixels) at this level. The HSD algorithm is better than the SMF algorithm and worse than the ACE algorithm in terms of the precision and recall values for this dataset.

Figure 6.47: HSD score image with 1.169 threshold

Figure 6.48: HSD score image with 1.09 threshold

Superpixel-based target detection algorithms are applied to this dataset as described in Table 4.1 and Table 4.2 after superpixels are obtained by using boundary update based superpixel extraction method. There are three combinations except from using pixels for background modeling and matching to adapt the traditional target detection methods such as ACE, SMF, and SAM to superpixel based detection methods. Target detection results which are obtained by using these methods strongly depend on the quality of the superpixel extraction algorithm. Because a superpixel region is represented with only one vector which is mean, medoid or centroid of the superpixel. Means are chosen as the representative of superpixels because of the better results. Spectral distance weight is chosen higher than the spatial distance weight for the superpixel extraction due to the importance of spectral similarity within the superpixels.

The best results of superpixel based target detection algorithms are given below.

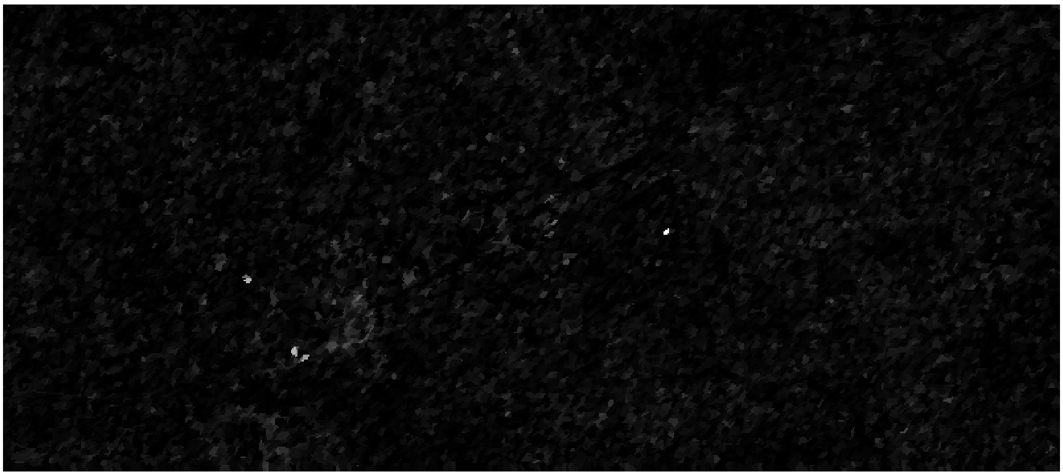

Figure 6.49: Superpixel adapted ACE score image - background modeling by pixels and matching with superpixel means

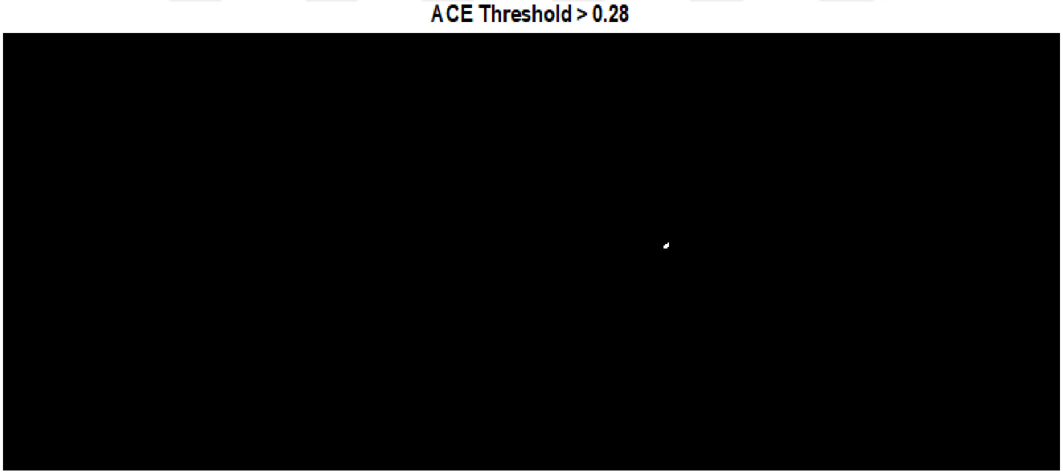

Figure 6.50: Superpixel adapted ACE thresholded score image - background modeling by pixels and matching with superpixel means

This combination gives scores for each superpixel because the matching operation is applied to superpixel representatives. On the other hand, all image pixels are used to calculate covariance matrix. The superpixel which includes full-target pixel has the highest score. However, this superpixel area contains totally 11 pixels and 8 target pixels among them. Therefore, the precision value is 0.73 (8 true positive pixels out of 11 pixels) and the recall value is 0.89 (8 true positive pixels out of 9 pixels). This result is quite impressive comparing to previous results from other algorithms. Because both of the performance metrics are high which is a desired result.

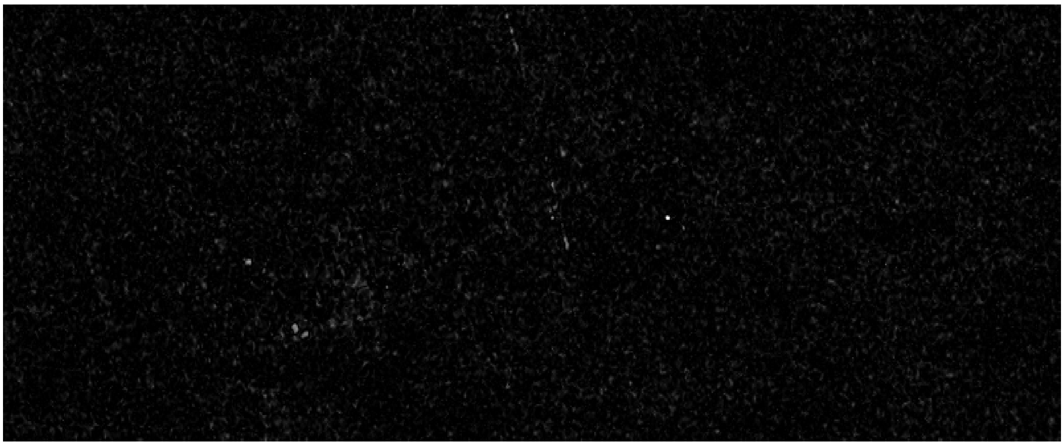

Figure 6.51: Superpixel adapted ACE score image - background modeling by superpixel means and matching with pixels

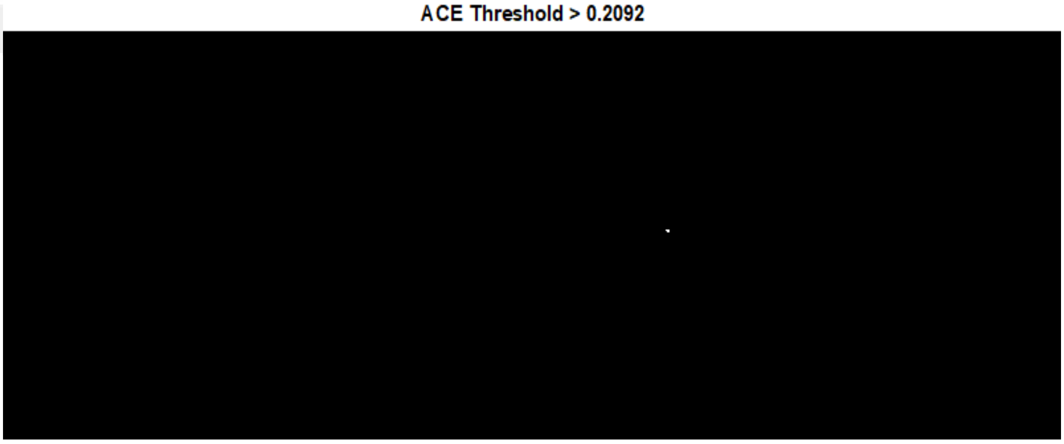

Figure 6.52: Superpixel adapted ACE thresholded score image - background modeling by superpixel means and matching with pixels

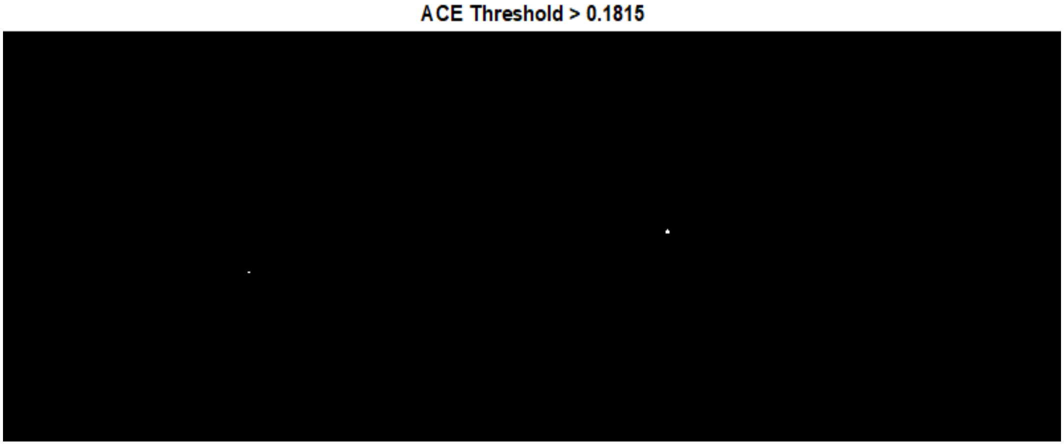

Figure 6.53: Superpixel adapted ACE thresholded score image - background modeling by superpixel means and matching with pixels

This combination is opposite the previous one. Superpixels are used to calculate covariance matrix and matching is applied to individual pixels. For this reason, each pixel has own score. The precision is 1 (5 true positive pixels out of 5 pixels) and the recall value is 0.56 (5 true positive pixels out of 9 pixels) for the first threshold value. If the second threshold value is applied, then the precision decreases to 0.78 (7 true positive pixels out of 9 pixels) and the recall increases to 0.78 (7 true positive pixels out of 9 pixels).

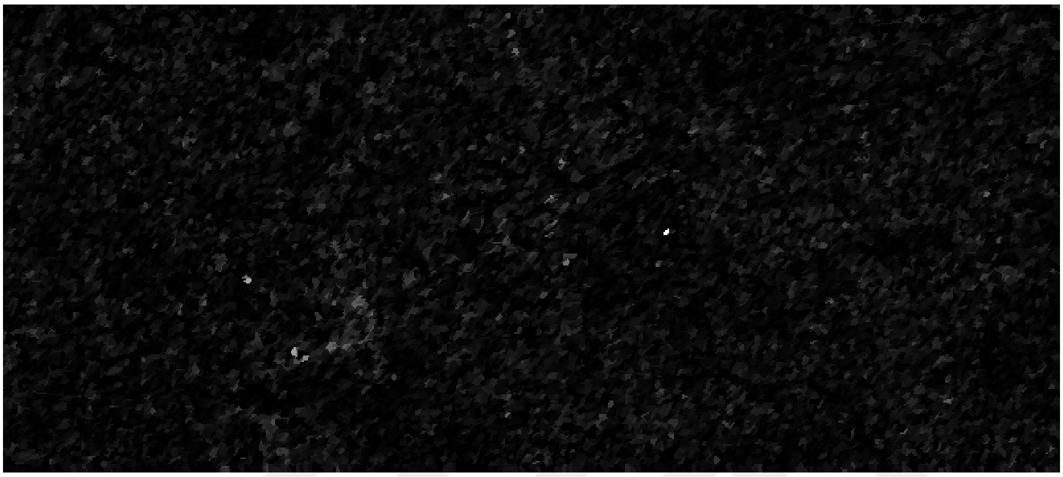

Figure 6.54: Superpixel adapted ACE score image - background modeling by superpixel means and matching with superpixel means

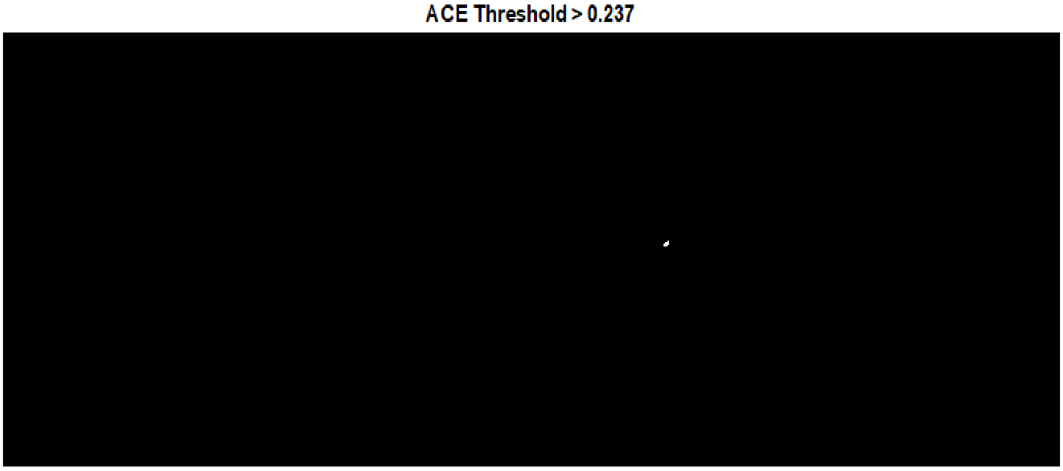

Figure 6.55: Superpixel adapted ACE thresholded score image - background modeling by superpixel means and matching with superpixel means

The results of this combination are same as the first combination which applies background modeling by pixels and matching with superpixel means. The only difference between two combinations is calculation of background covariance matrix. According to these results for this dataset, background modeling with pixels or superpixel means does not affect the detection scores when the matching operation is applied to superpixel means. The average size of the superpixels is too less, so the calculation of the covariance matrix is almost similar to the first combination. Therefore, both combinations give similar results.

Superpixel based hybrid detectors have an extra option which is abundance estimation. Therefore, totally seven combinations can be applied except from using pixels for background modeling, matching, and abundance estimation. Firstly, abundance estimation is obtained by using pixels or superpixel representatives. Then, the abundance estimation values are used in detection algorithm as explained in the Section 2.4.2 and Section 2.4.3. The best results of the combinations are given below.

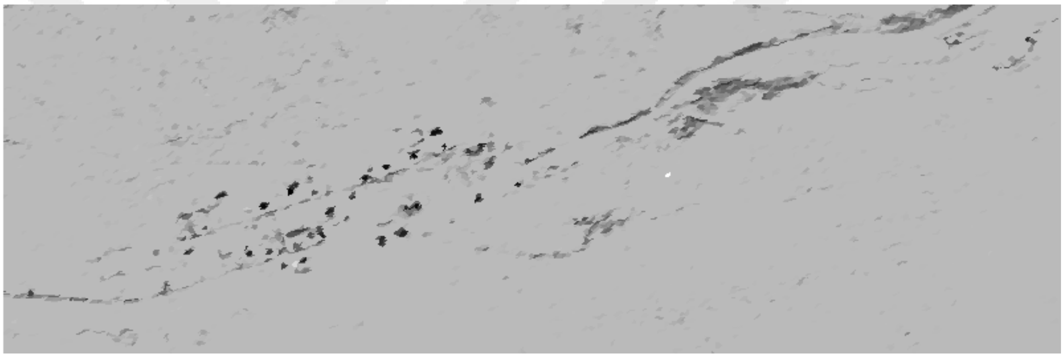

Figure 6.56: Superpixel adapted HSD score image - background modeling by pixels, matching with superpixel means and abundance estimation by using superpixels

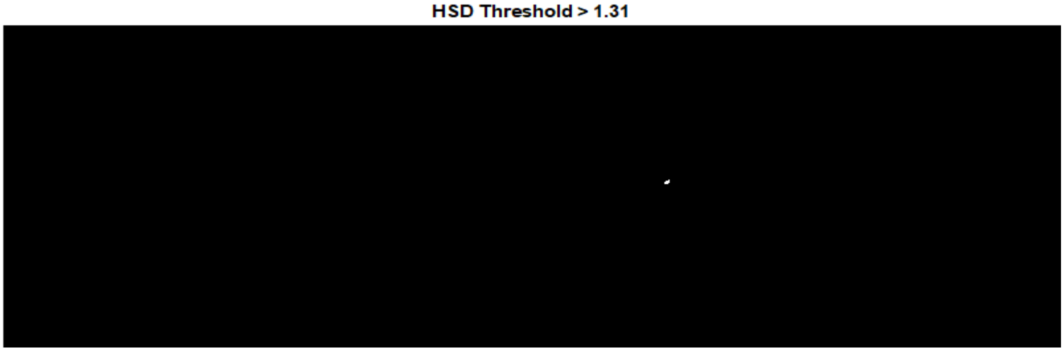

Figure 6.57: Superpixel adapted HSD thresholded score image - background modeling by pixels, matching with superpixel means and abundance estimation by using superpixels

The result of this algorithm is similar to superpixel adapted ACE-background modeling by pixels and matching with superpixel means. Moreover, the background is highly suppressed by using this algorithm. The target pixels easily discriminate from other pixels as it can be seen from Figure 6.56. The precision value is 0.73 (8 true positive pixels out of 11 pixels) and the recall value is 0.89 (8 true positive pixels out of 9 pixels) for given threshold value in Figure 6.57. The same result can be obtained by the superpixel adapted HSD algorithm which applies background modeling by superpixel means, matching with superpixel means and abundance estimation by using superpixels. It is expected, because as mentioned before background modeling by superpixels or pixels does not change the results when the other operations are applied with same combinations.

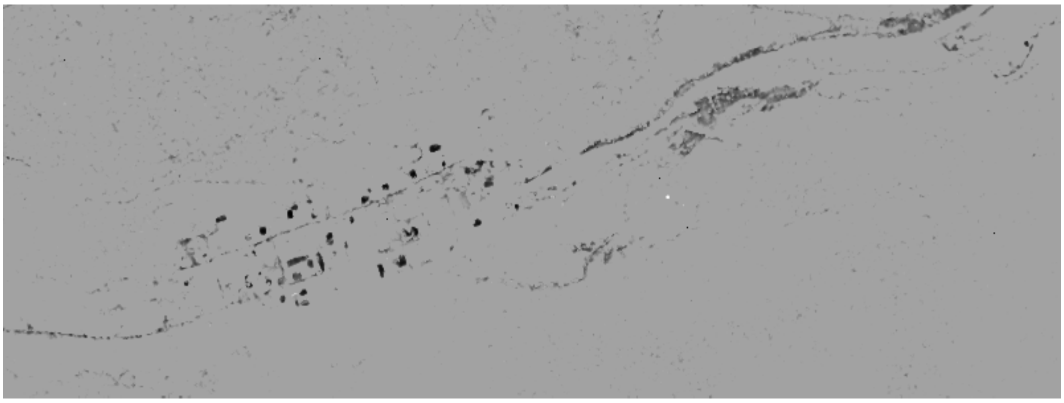

Figure 6.58: Superpixel adapted HSD score image - background modeling by superpixel means, matching with pixels and abundance estimation by using pixels

Figure 6.59: Superpixel adapted HSD thresholded score image - background modeling by superpixel means, matching with pixels and abundance estimation by using pixels

The result of this algorithm is not as good as previous algorithm. However, the precision value is 0.7 (7 true positive pixels out of 10 pixels) and the recall value is 0.78 (7 true positive pixels out of 9 pixels) in the result shown in Figure 6.59.

The overall performance of the superpixel adapted target detection algorithms is quite satisfactory. The main reasons of these results are the accurate extraction of superpixels by considering the spectral similarity between pixels, choosing right superpixel representatives which contain spectral signature of the corresponding superpixel regions and successful application of the target detection algorithms. All of the results for the given algorithms are shown in Table 6.1. On the other hand, after the superpixels are extracted, performing both of the background modeling and matching operations are faster than the baseline methods due to the number of superpixels.

Algorithm	**Precision**	**Recall**
SMF	0.076	1
ACE	0.16	1
HSD	0.167	0.44
SP-ACE (Background Modeling:Pixels Matching:SP)	0.73	0.89
SP-ACE (Background Modeling:SP Matching:SP)	0.73	0.89
SP-ACE (Background Modeling:SP Matching:Pixels)	0.78	0.78
SP-HSD(Background Modeling:SP Matching:Pixels Abundance:Pixels)	0.7	0.78
SP-HSD(Background Modeling:Pixels Matching:SP Abundance:SP)	0.73	0.89
SP-HSD(Background Modeling:SP Matching:SP Abundance:SP)	0.73	0.89

Table 6.1: The performance metric results of target detection methods for RIT dataset

Superpixel based target detection algorithms are also tested on prepared datasets shown in Figure 5.11 and Figure 5.14. These datasets are obtained by experimental setup which is explained in Section 5.1. The aim of preparing these datasets is testing the algorithms for more challenging problems such as the detection of mixture pixels. The solutions of materials are homogeneously sprayed on the metal plates to create mixture pixels. The spectral signatures of target materials and metal plates are obtained by using ASD spectroradiometer. All of these spectral signatures are given in Figure 6.60-6.66.

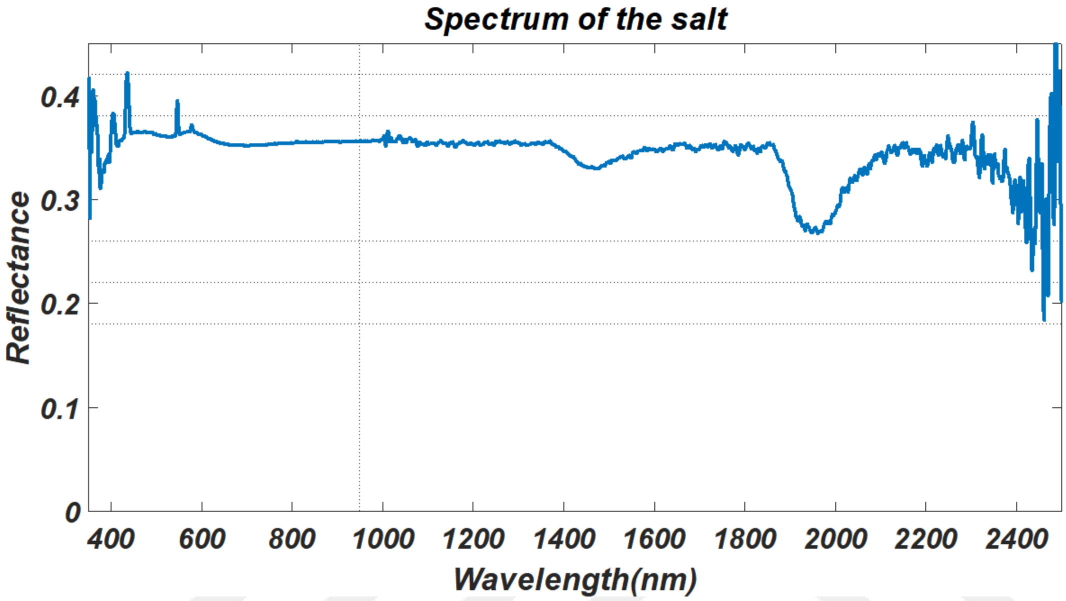

Figure 6.60: The spectral signature of the salt

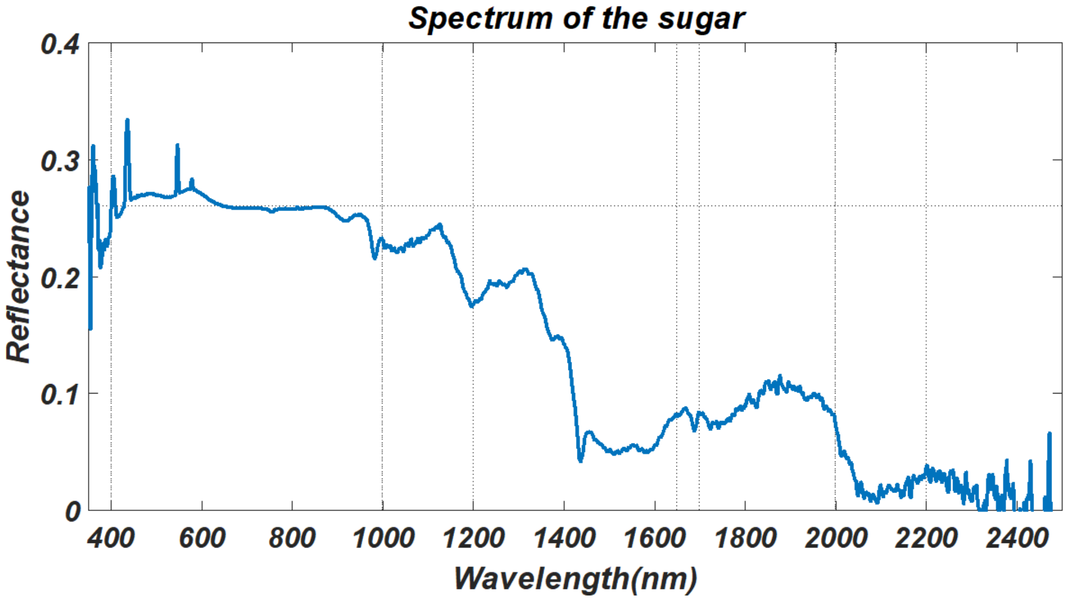

Figure 6.61: The spectral signature of the sugar

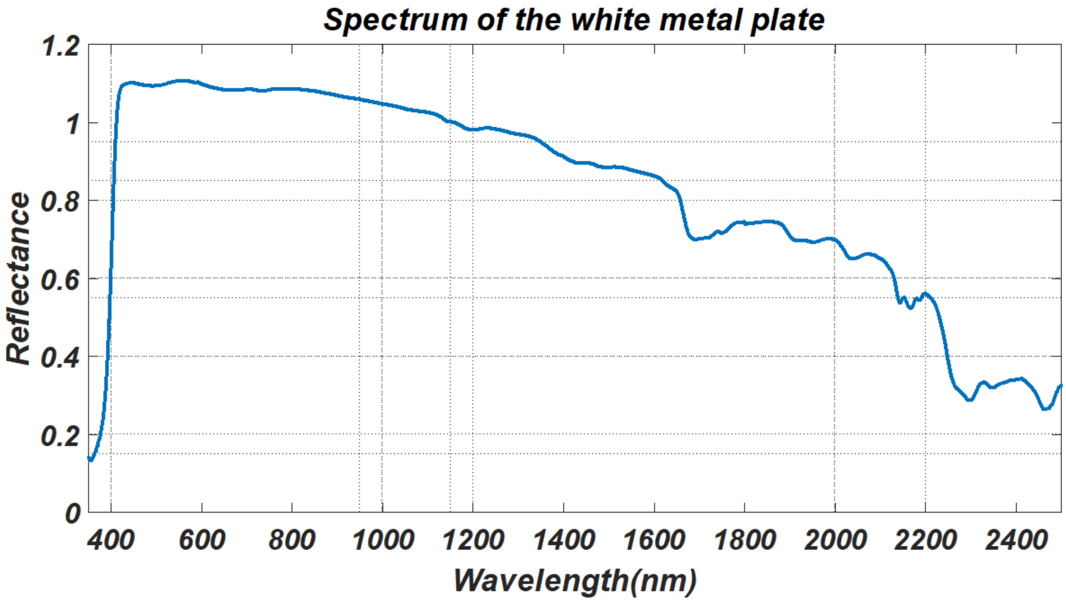

Figure 6.62: The spectral signature of the white metal plate

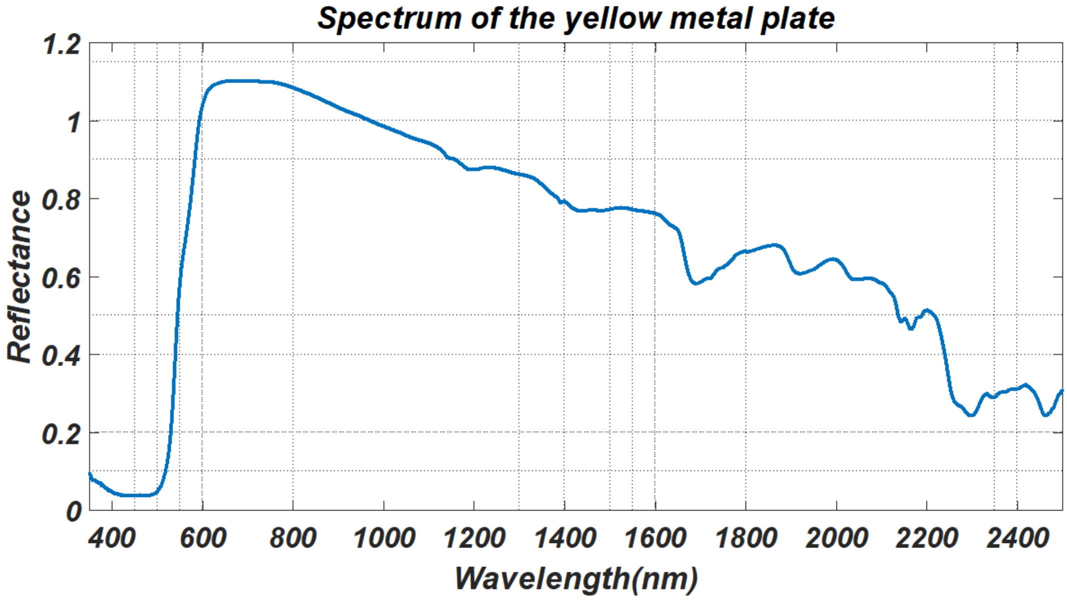

Figure 6.63: The spectral signature of the yellow metal plate

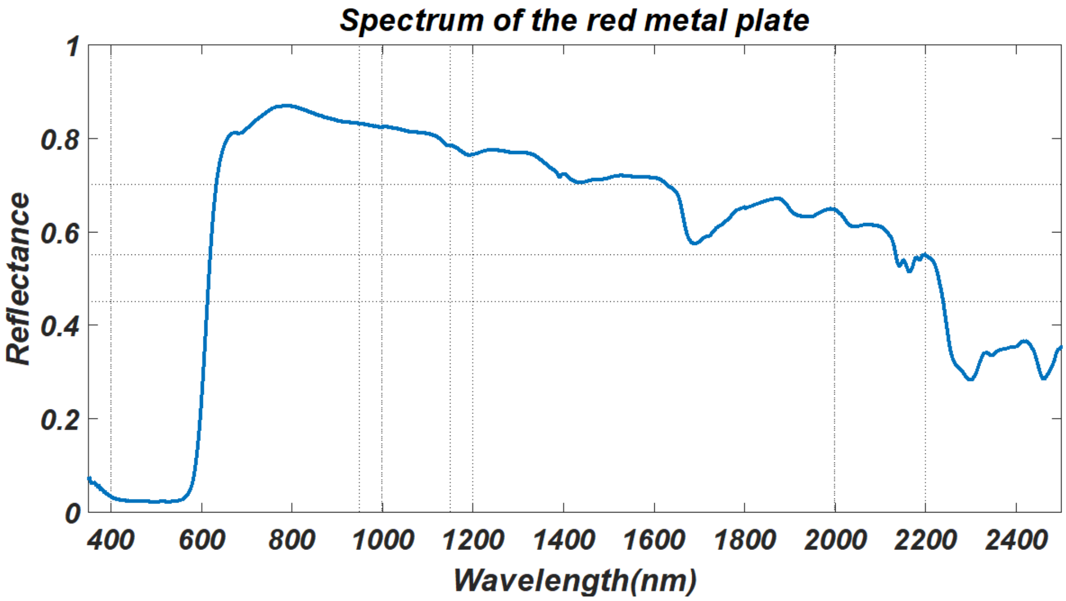

Figure 6.64: The spectral signature of the red metal plate

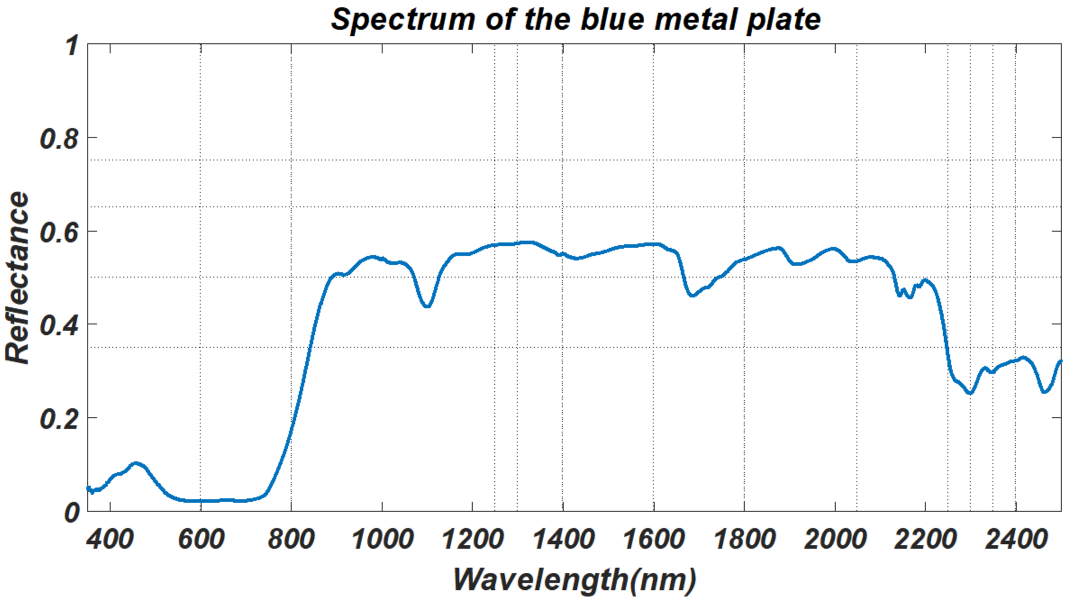

Figure 6.65: The spectral signature of the blue metal plate

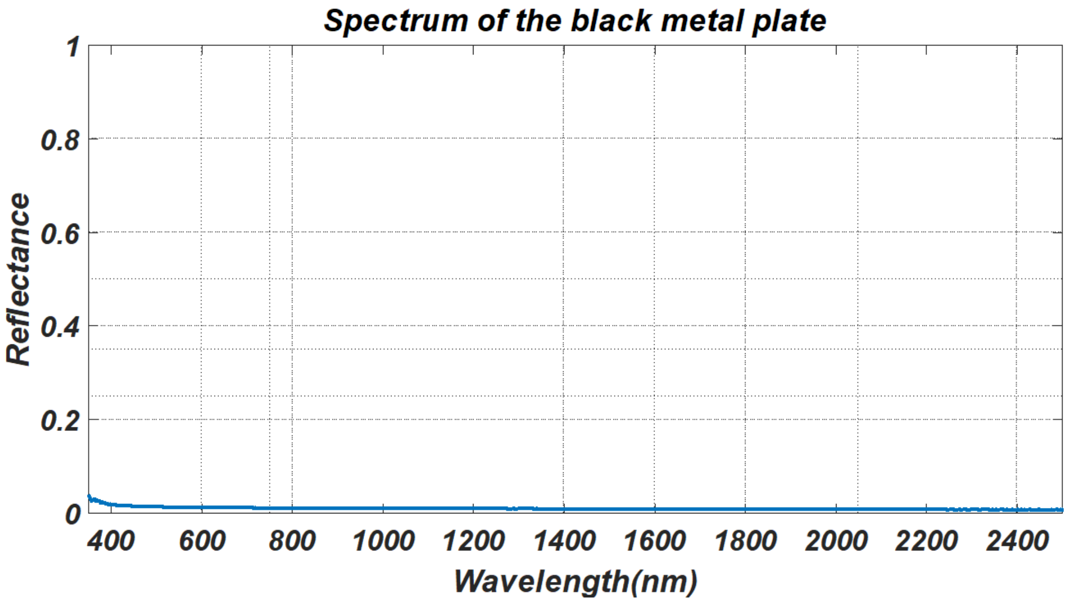

Figure 6.66: The spectral signature of the black metal plate

The specific spectral characteristics of the target materials which are sugar and salt can be observed in the interval of 1000-1700 nm. There are local minimum curves in the interval of 1400-1700 nm for these targets. White and yellow metal plates have reflectance values which are greater than 1 in some wavelengths. The main reason of this situation is the high-reflective characteristic of the metal plate comparing to the spectralon.

The datasets shown in Figure 5.11 and Figure 5.14 include 71 spectral bands in the interval of 1000-1700 nm. Because the specific characteristics of the target materials in the interval of 1400-1700 nm, 31 spectral bands which are in this interval are used for the target detection. The superpixel number for this dataset is chosen as 5000. The best results of the target detection algorithms for outdoor dataset shown in Figure 5.11 are given below as score images and precision-recall curves.

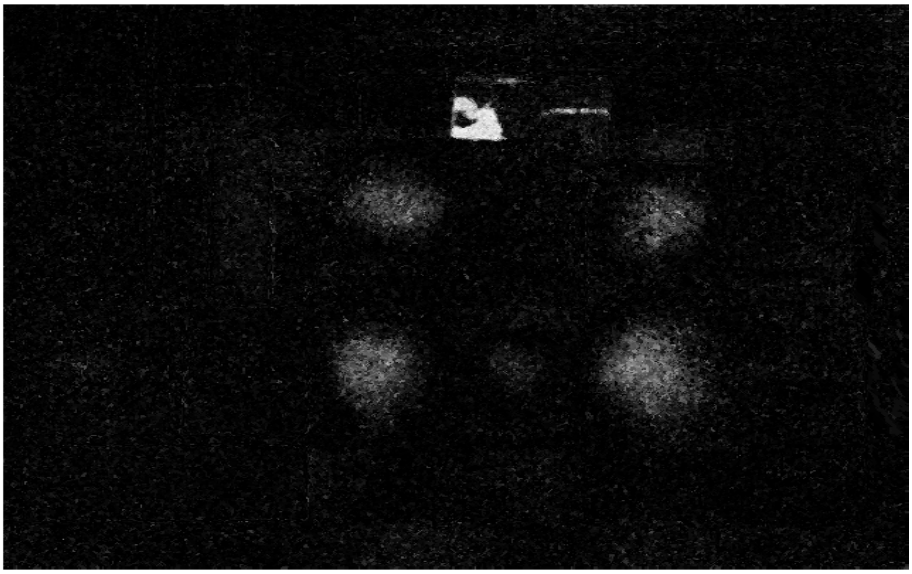

Figure 6.67: ACE score image for outdoor dataset

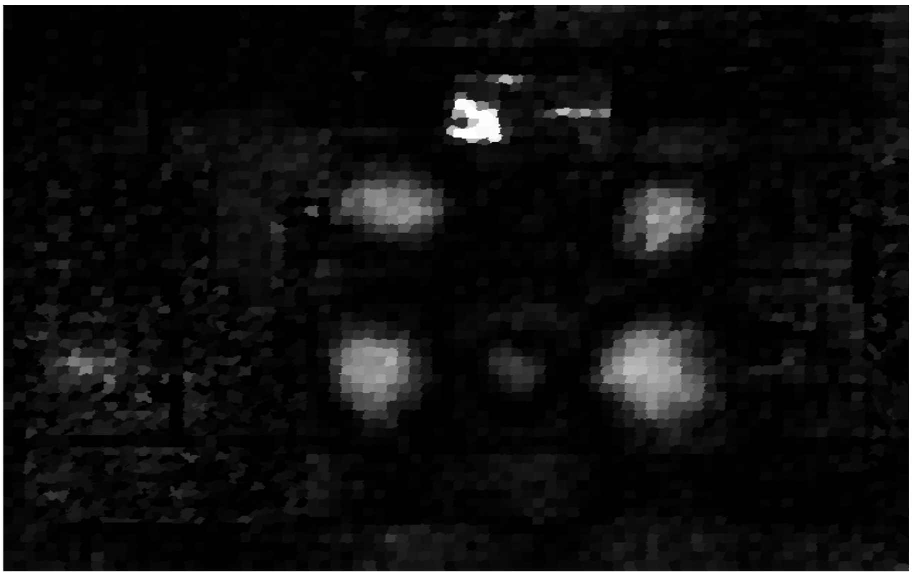

Figure 6.68: Superpixel adapted ACE score image - background modeling by pixels and matching with superpixel means

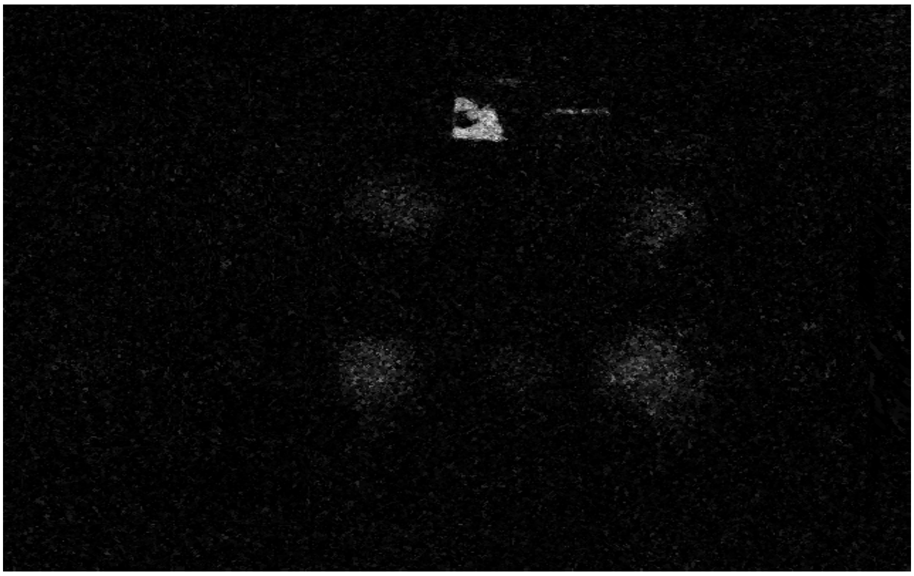

Figure 6.69: Superpixel adapted ACE score image - background modeling by superpixel means and matching with pixels

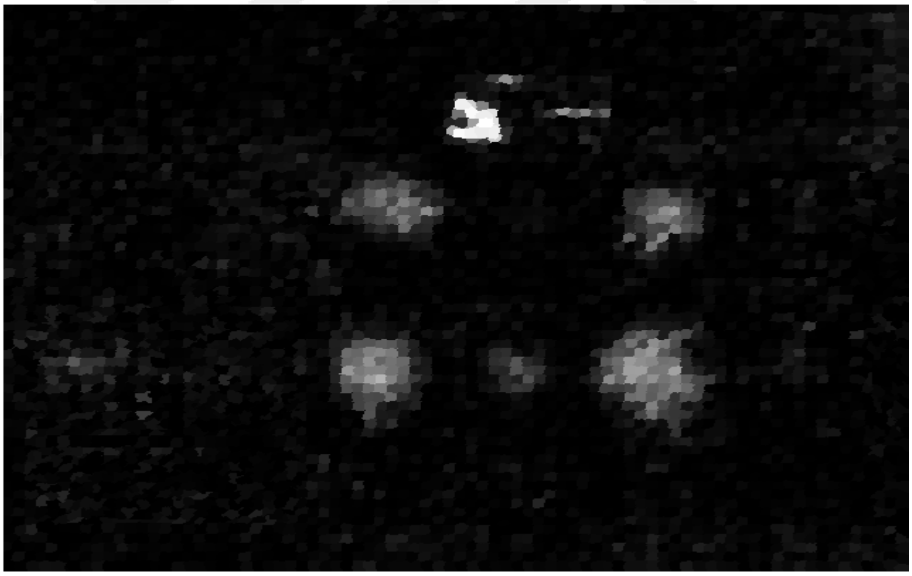

Figure 6.70: Superpixel adapted ACE score image - background modeling by superpixel means and matching with superpixel means

The precision-recall curves of the superpixel-based ACE algorithm are given in Figure 6.71. To compare the performance results of the algorithms, the average precision can be utilized. The average precision is defined as the area under the curve.

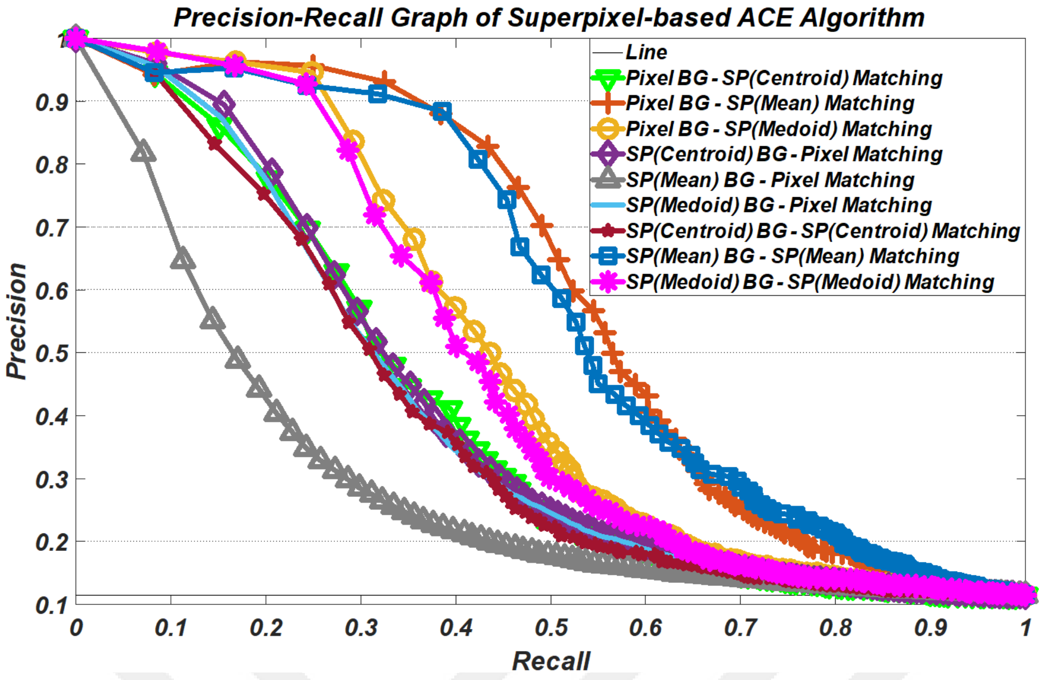

Figure 6.71: Precision-recall curves of the superpixel-based ACE algorithm

Figure 6.71 shows the comparison of the superpixel representatives of the ACE algorithm. Selection of the superpixel representative affects the performance. The best representatives of the superpixels are the mean values of the superpixels for this dataset. As a preprocessing step, a 3x3 spectral median filter is applied to the image pixels for eliminating the noisy pixels and dead pixels in the image. Therefore, the mean values of the superpixels can be expected as the best representatives due to the extinction of outlier elements of the superpixels.

There is another important outcome can be commented from the precision-recall curve. The precision-recall curve is expected to behave monotonically decreasing. However, if there are small errors on the ground truth or the highest scores are obtained from the false positive pixels, this behaviour cannot be observed in some situations such as the blue line that is shown in the Figure 6.71. To analyze this situation, the number of samples is increased for the ACE algorithm that uses superpixels for background modeling and matching. The precision-recall curve of this algorithm with points on the line is given in Figure 6.72.

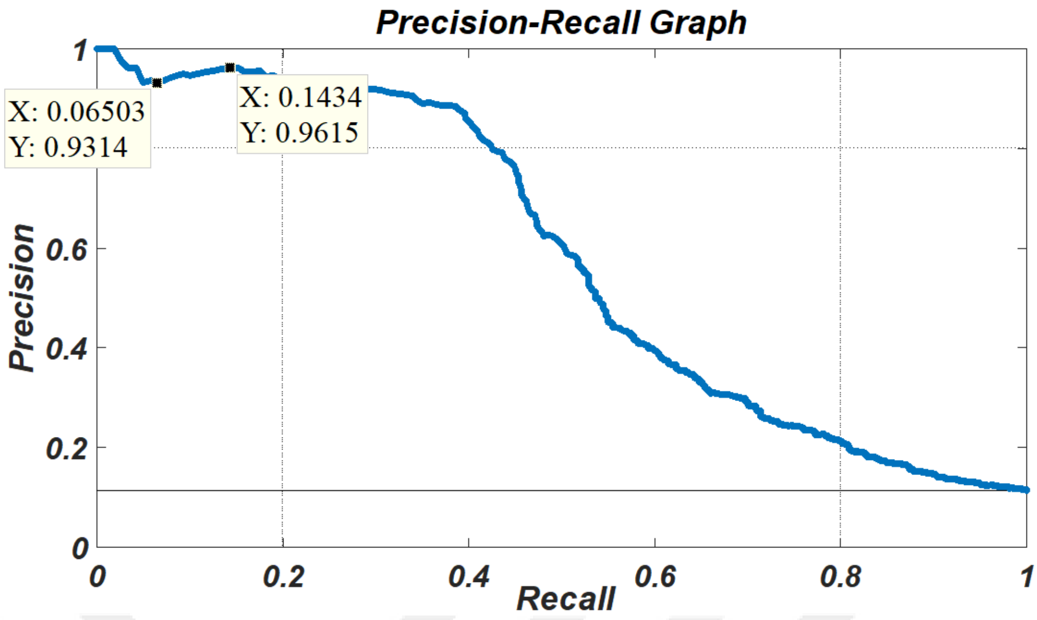

Figure 6.72: Precision-recall curves of the ACE algorithm that uses superpixels for background modeling and matching

Two threshold levels are chosen for comparison. The recall values are 0.065 and 0.14 and the corresponding precision values are 0.93 and 0.96. The precision values are not monotonically decreasing in contrast to expected behavior. Therefore, the score images of these thresholds which are given in Figure 6.73 and 6.74 have to be analyzed.

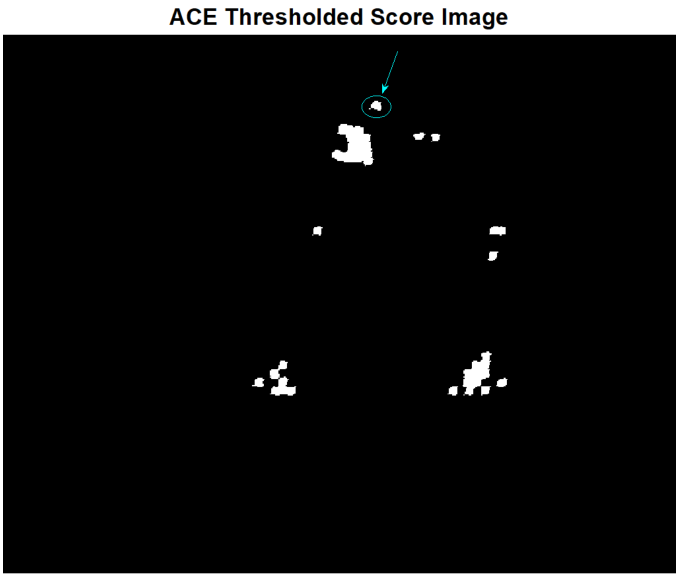

Figure 6.73: ACE thresholded score image (precision=0.93 and recall=0.065)

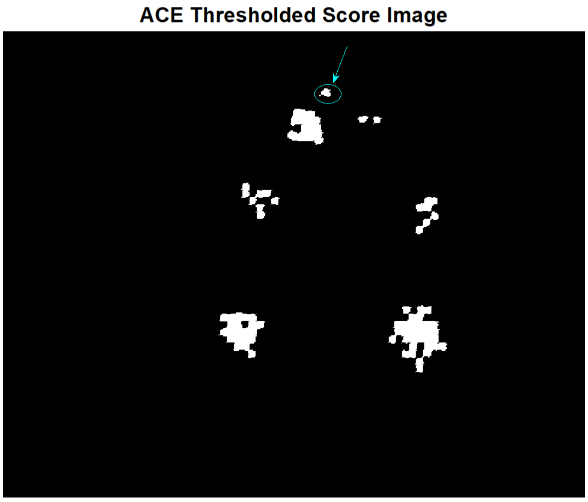

Figure 6.74: ACE thresholded score image (precision=0.96 and recall=0.14)

The region which is shown by an arrow is not included to ground truth. Although this region includes the target material, it was not drawn in the ground truth. Also, the score of this region is higher than some of the ground truth regions. Therefore, the system percepts this region as the false positive region. As a result, the line between the two selected points does not show the monotonically decreasing behavior.

The precision-recall curves of the score images can be obtained by using the ground truth of this dataset. The mean values of the superpixels are chosen as the representatives because of the performance results which are given in Figure 6.71. The results of four combinations of the ACE algorithm are illustrated in Figure 6.75.

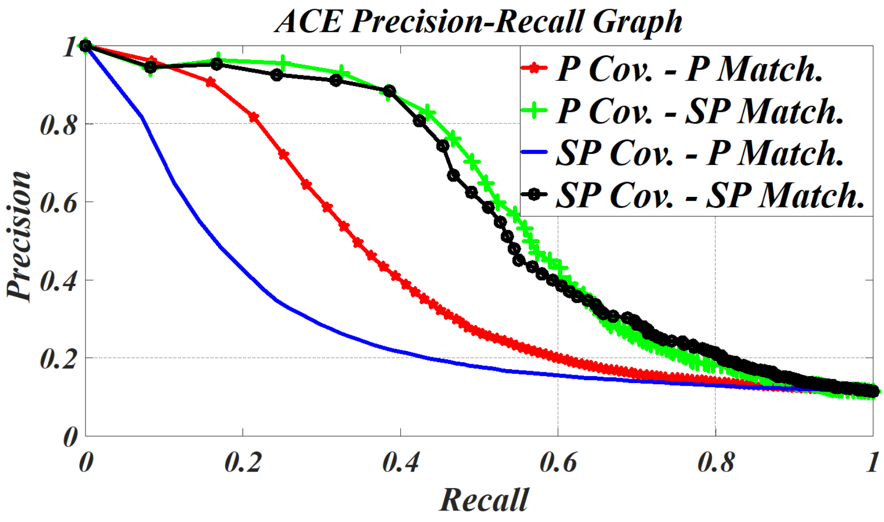

Figure 6.75: Precision-recall curves of the ACE algorithm

As it can be inferred from the results, the best algorithm is the superpixel adapted ACE algorithm which uses pixels for background modeling and superpixels for matching. The obtained covariance matrix by using pixels can model the variations of the data better than superpixels. On the other hand, main positive effect in the performance can be provided by using superpixel representative for matching operation.

Another successful algorithm for the outdoor dataset is DTDCA. The best results of the DTDCA algorithm are given below as score images and precision-recall curves.

Figure 6.76: DTDCA score image for outdoor dataset

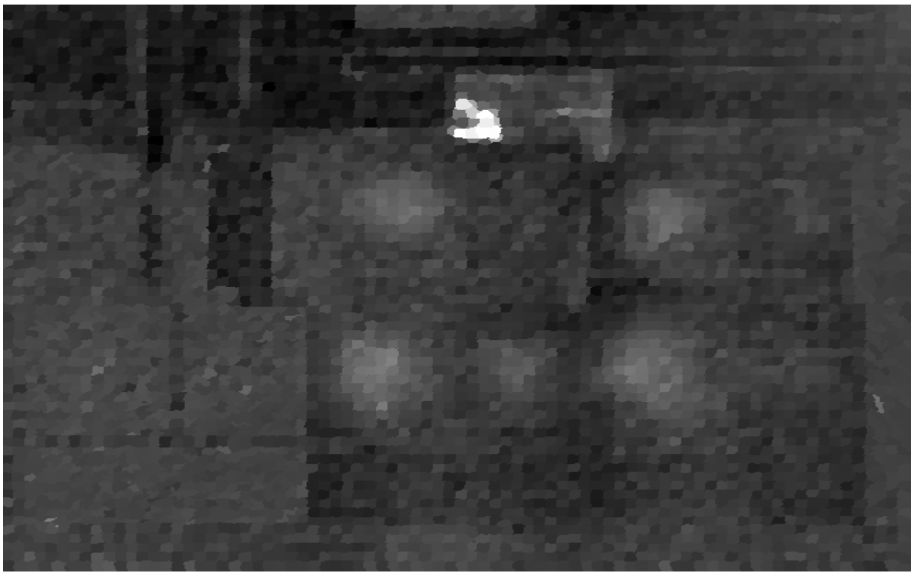

Figure 6.77: Superpixel adapted DTDCA score image - background modeling by pixels and matching with superpixel means

Figure 6.78: Superpixel adapted DTDCA score image - background modeling by superpixel means and matching with pixels

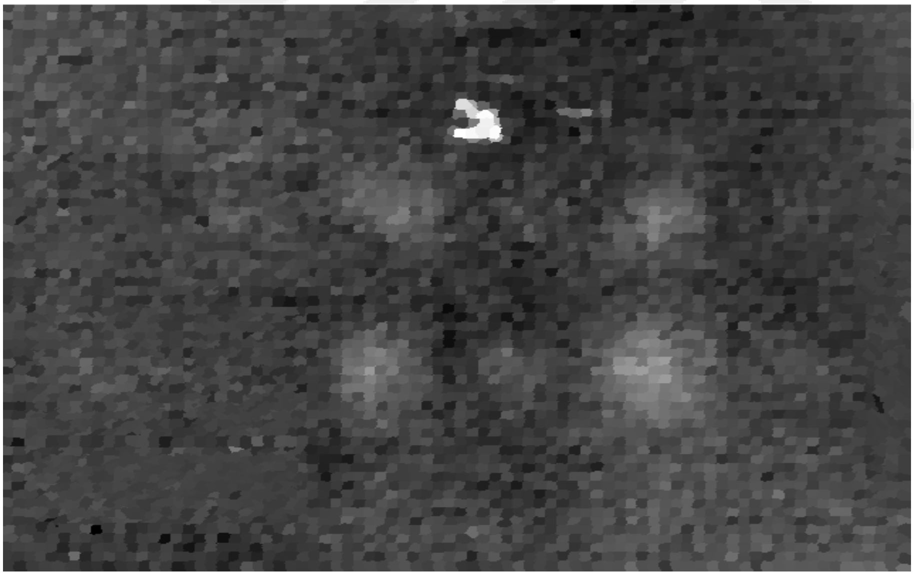

Figure 6.79: Superpixel adapted DTDCA score image - background modeling by superpixel means and matching with superpixel means

Figure 6.80 shows the comparison of the superpixel representatives of the DTDCA algorithm. The similar results which are compatible with the ACE algorithm are obtained due to the same reasons.

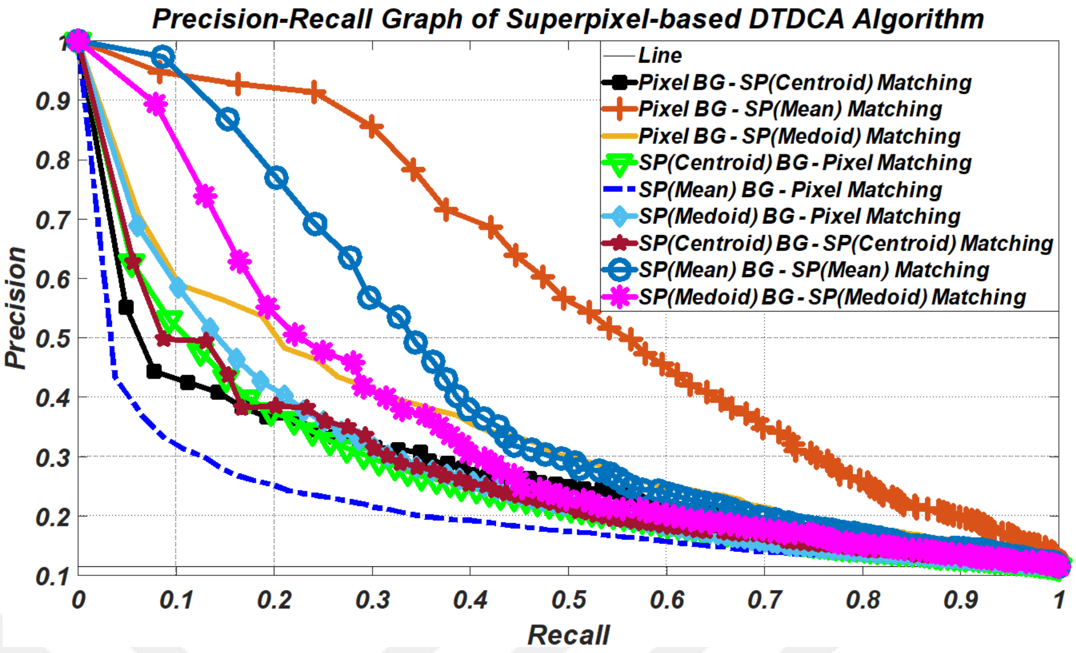

Figure 6.80: Precision-recall curves of the superpixel-based DTDCA algorithm

The mean values of the superpixels are chosen as the representatives because of the performance results which are given in Figure 6.80. Figure 6.81 shows the performance results of four combinations of the DTDCA algorithm in terms of precision and recall values.

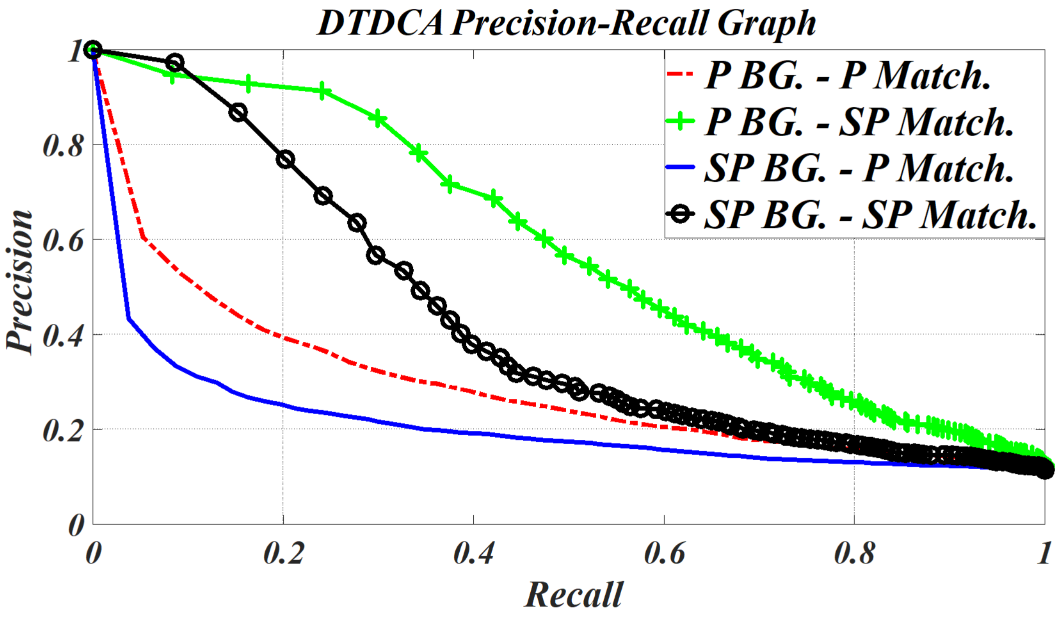

Figure 6.81: Precision-recall curves of the DTDCA algorithm

The results are similar to the ACE algorithm. The best combination of the DTDCA algorithm is performing background modeling by pixels and matching with superpixels. Using superpixels for the target detection increases the performance. The only exception is the combination which models the background by using superpixels and performs matching by using pixels. Creating background covariance matrix by using superpixels and matching with pixels is logically incorrect. Because modeling background with the set which has fewer elements and applying this model to the set which has more elements creates insufficient results.

As the last algorithm for the outdoor dataset, the results of the HUD algorithm are quite satisfactory. The target detection results of the HUD algorithm in terms of precision and recall metrics are given below. Also the best results of the combinations are given as score images.

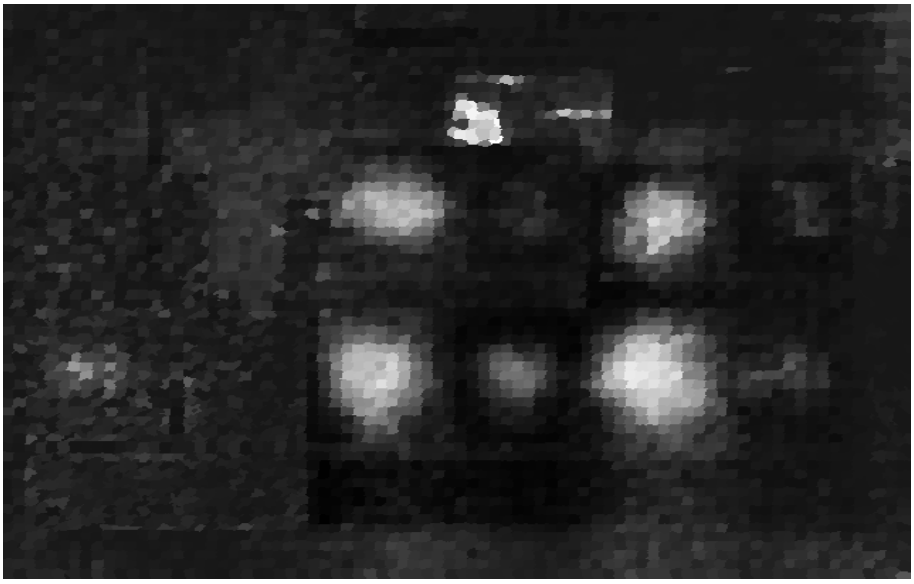

Figure 6.82: Superpixel adapted HUD score image - background modeling by pixels, matching with superpixels and abundance estimation by using superpixels

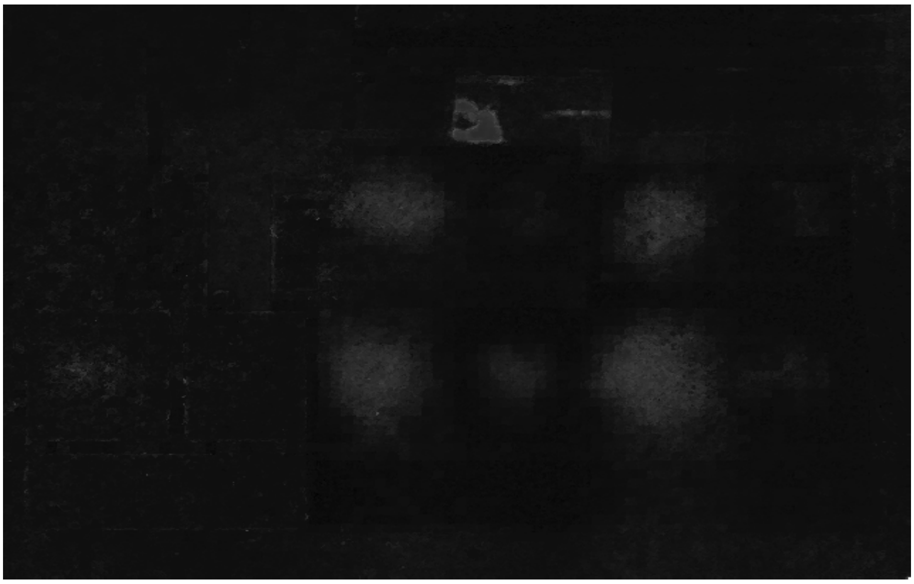

Figure 6.83: Superpixel adapted HUD score image - background modeling by pixels, matching with superpixels and abundance estimation by using pixels

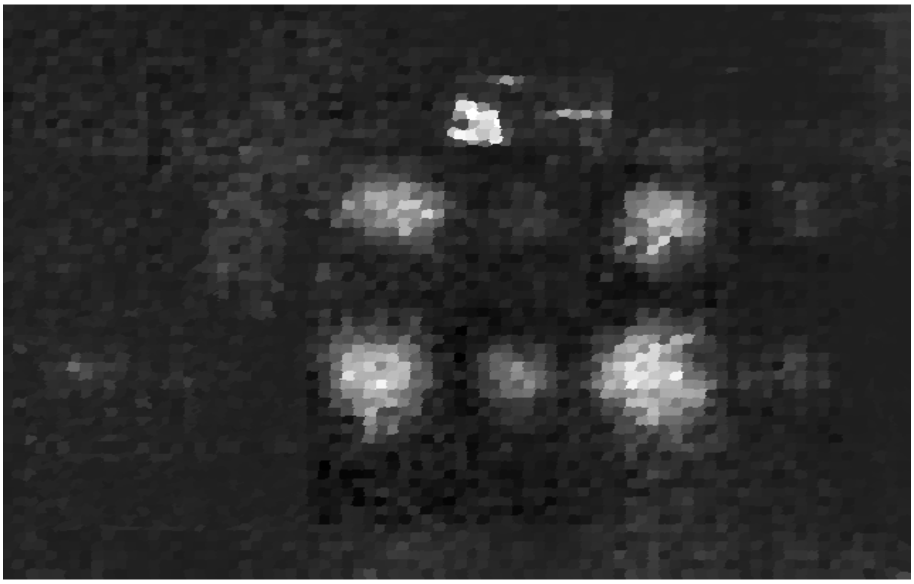

Figure 6.84: Superpixel adapted HUD score image - background modeling by superpixels, matching with superpixels and abundance estimation by using superpixels

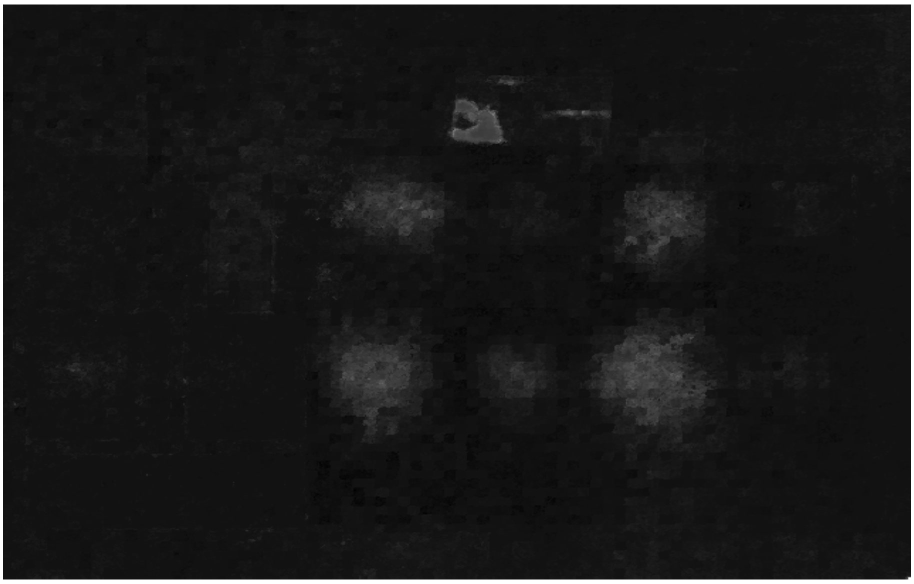

Figure 6.85: Superpixel adapted HUD score image - background modeling by superpixels, matching with superpixels and abundance estimation by using pixels

Figure 6.86 illustrates the performance results of eight combinations of the HUD algorithm in terms of precision and recall values.

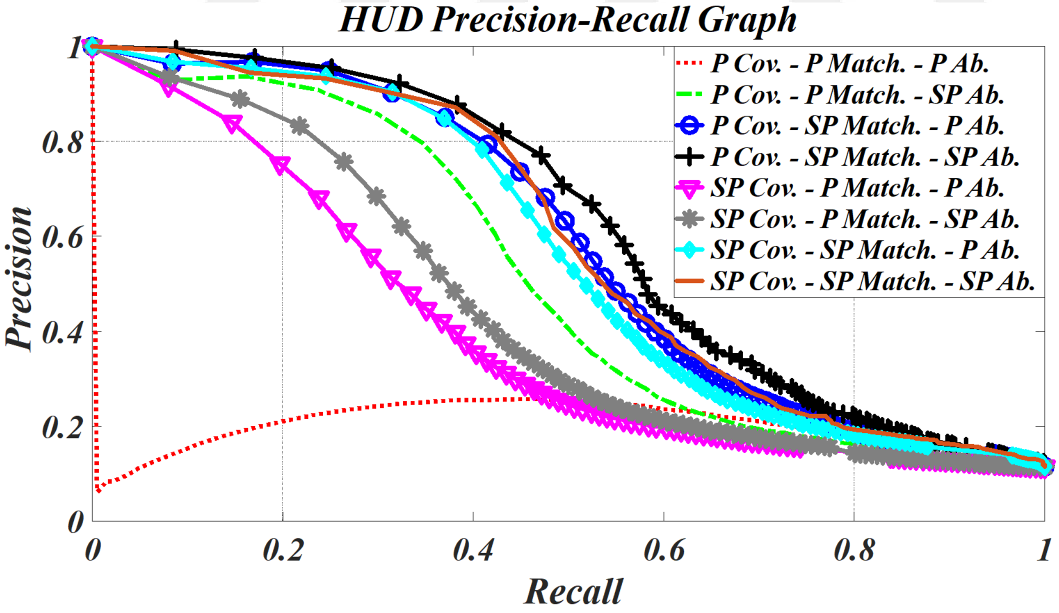

Figure 6.86: Precision-recall curves of the HUD algorithm

The results show that modeling background by using pixels and matching with superpixels gives the best performances. Besides this information, the abundance estimation by using superpixels increase the detection performance comparing to abundance estimation with pixels. These results which are obtained from three different target detection algorithms reveal that calculation of the covariance matrix by using pixels which have more elements than the set of superpixels and performing the matching operation with the superpixel representatives which provide the suppression of the noise effects are good combinations for the aim of target detection. Figure 6.87 shows the comparison of these three algorithms in terms of their best combinations.

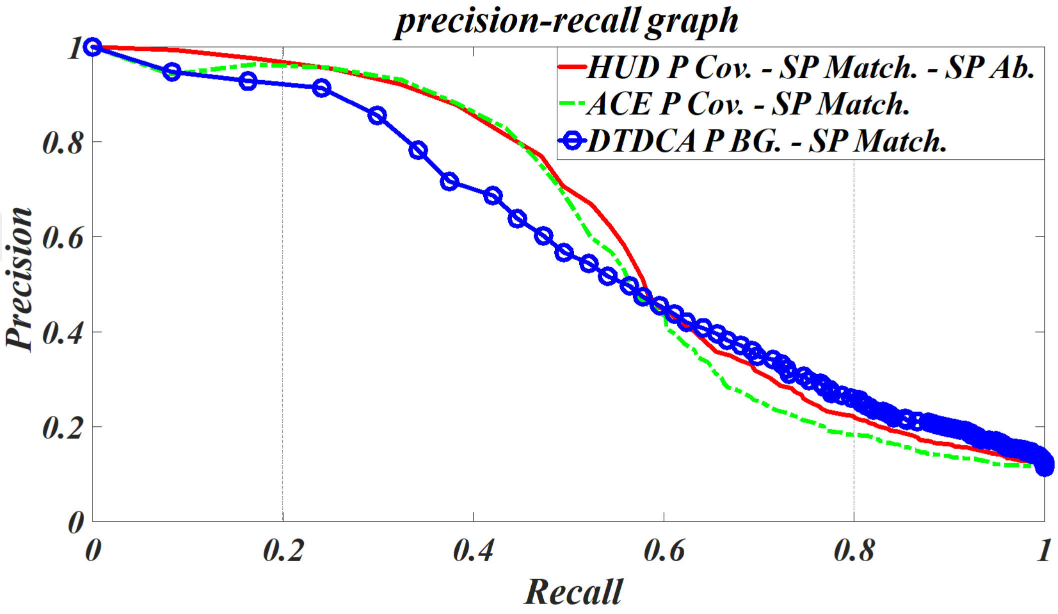

Figure 6.87: Precision-recall curves of the best combinations of the ACE, DTDCA and HUD algorithms

The best performance of these three algorithms in terms of average precision is obtained by using the HUD algorithm. Application of linear unmixing with the baseline target detection methods in hybrid algorithms positively affects the performance results. On the other hand, the superpixel adapted ACE algorithm shows the performance as good as the HUD algorithm. The performances of the background modeling of these algorithms are quite satisfactory. Also using superpixel representatives for target detection provides better precision-recall values due to the inclusion of the spatial relations between pixels.

The dataset which is shown in Figure 5.14 is used to test target detection algorithms. The best results of the target detection algorithms for the salt are given in Figure 6.88, 6.89 and 6.90 as score images and precision-recall curves.

Figure 6.88: Superpixel adapted ACE score image - background modeling by pixels, matching with superpixels

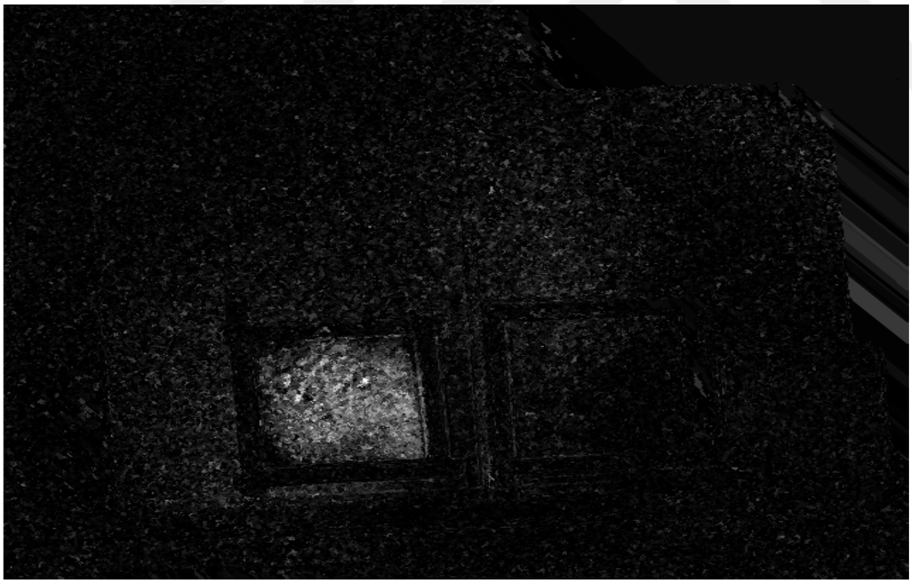

Figure 6.89: Superpixel adapted ACE score image - background modeling by pixels, matching with pixels

Figure 6.90 illustrates the performance results of four combinations of the ACE algorithm in terms of precision and recall values.

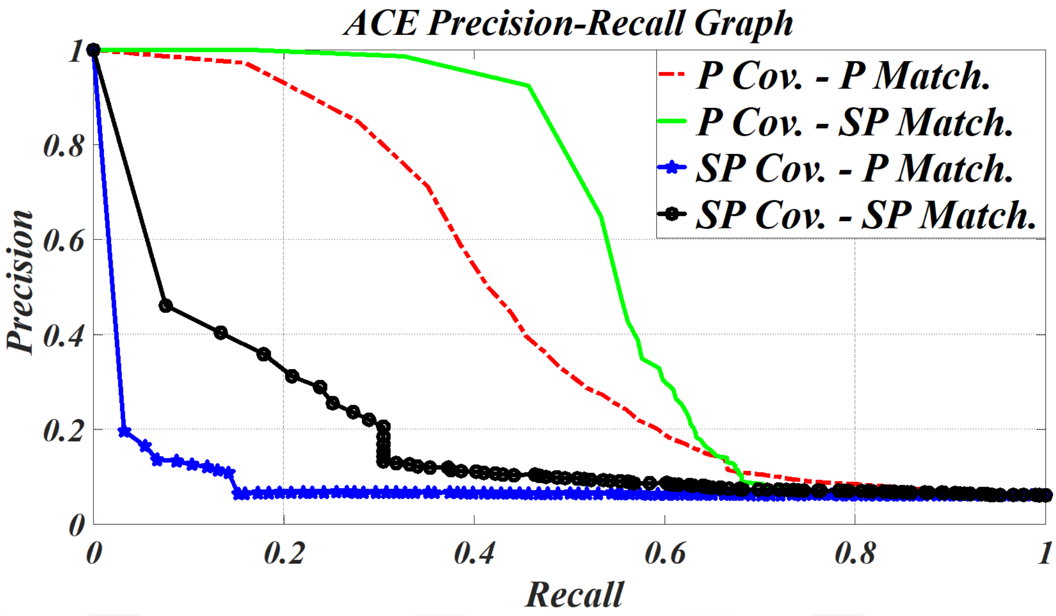

Figure 6.90: Precision-recall curves of the ACE algorithm

As in the previous results, the best algorithm models the background by using pixels and performs matching by superpixels. The main difference from the previous result is the performance of the method that models the background by using superpixels performs matching with superpixels. One of the main reasons of this situation can be the inappropriate background modeling by the superpixels in the ACE algorithm due to existing shadows and the non-homogeneously spreading of the light on the scene.

Figure 6.91: Superpixel adapted DTDCA score image - background modeling by superpixels, matching with superpixels

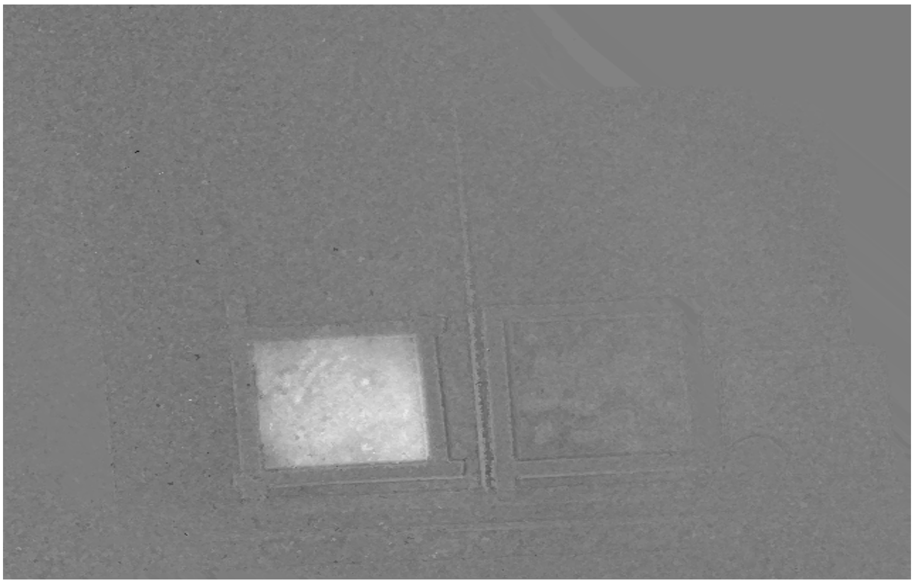

Figure 6.92: Superpixel adapted DTDCA score image - background modeling by superpixels, matching with pixels

Figure 6.93 illustrates the performance results of four combinations of the DTDCA algorithm in terms of precision and recall values.

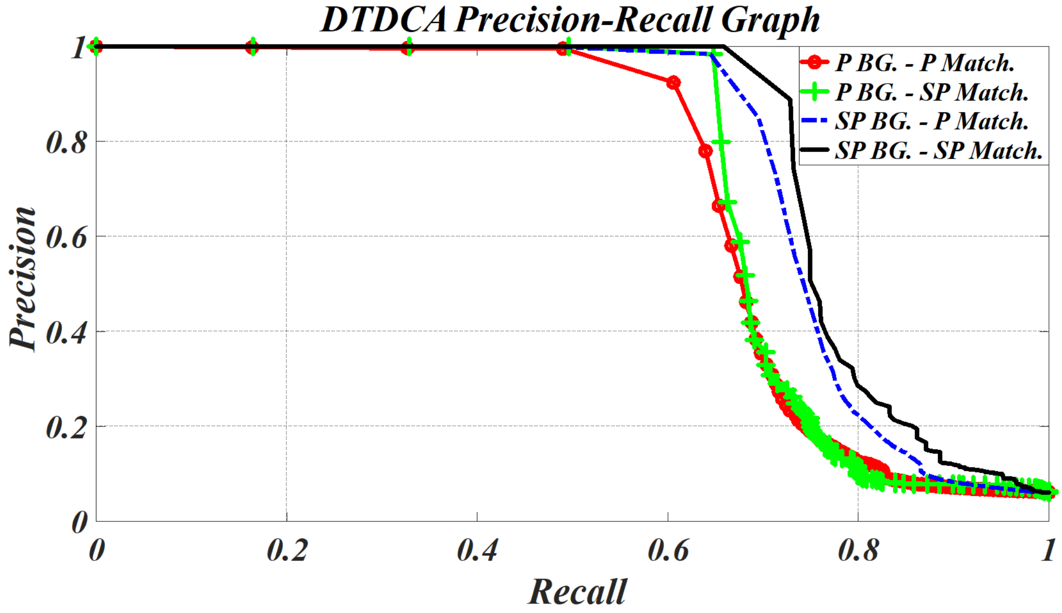

Figure 6.93: Precision-recall curves of the DTDCA algorithm

Although the best combination of this algorithm is using superpixels for both operations, the results of the other combinations which include superpixels are close to the best combination. In contrast to the ACE algorithm, the background modeling by the superpixels in this algorithm for this dataset reveals better precision-recall curves. The background signatures are obtained by applying the subspace-projection to the target spectrum in this algorithm. For this reason, the more accurate model of the background can be obtained.

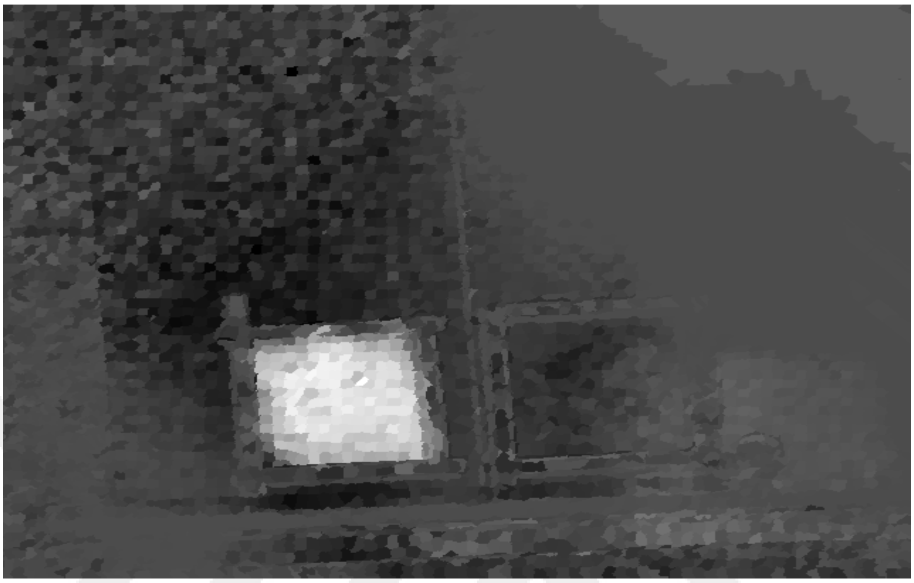

Figure 6.94: Superpixel adapted HUD score image - background modeling by pixels, matching with superpixels and abundance estimation by using superpixels

Figure 6.95: Superpixel adapted HUD score image - background modeling by pixels, matching with superpixels and abundance estimation by using pixels

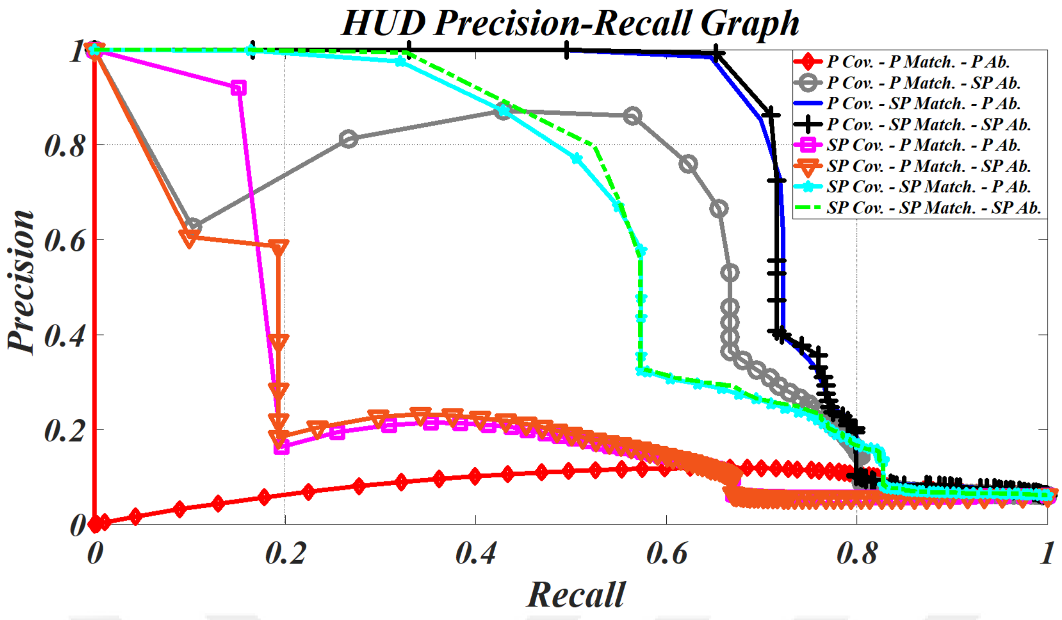

Figure 6.96: Precision-recall curves of the HUD algorithm

Figure 6.96 illustrates the performance results of eight combinations of the HUD algorithm in terms of precision and recall values. The results of these algorithms are compatible with the previous results of the HUD algorithms for the outdoor dataset. The HUD algorithm is as successful as the DTDCA algorithm for this dataset. Applying linear unmixing to the image scene with the combination of background modeling by pixels and matching with superpixels increases the performance. Figure 6.97 illustrates the comparison of the best combinations for three algorithms.

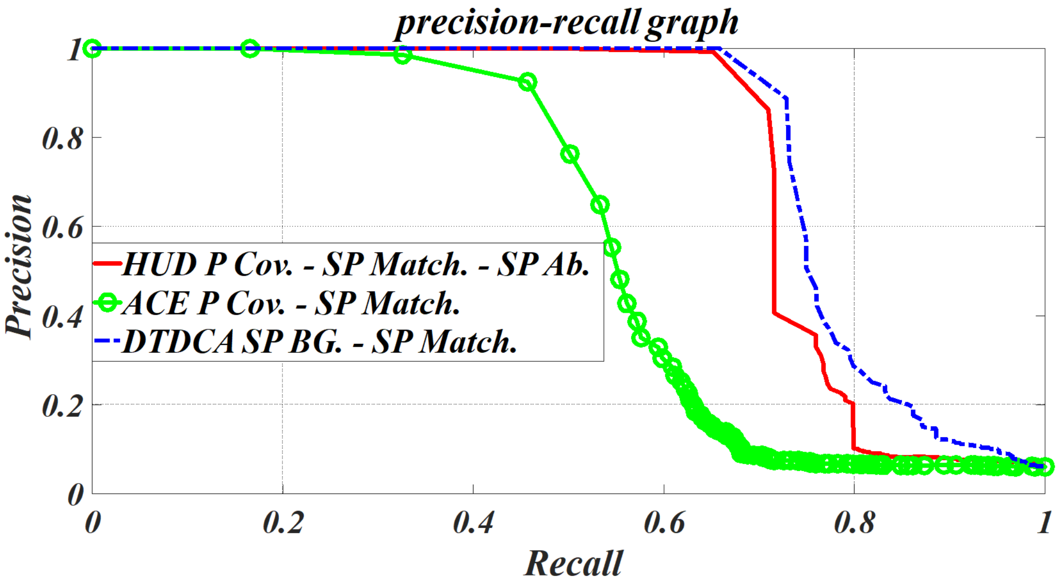

Figure 6.97: Precision-recall curves of the best combinations of the ACE, DTDCA and HUD algorithms

The DTDCA algorithm that models the background by using superpixels and performs matching operation with superpixels shows the best performance. However, the performance of the hybrid detector is as good as the DTDCA algorithm. As a result, these algorithms are more robust than the ACE algorithm for an unbalanced scene in terms of the light conditions. Another result can be extracted from these results is that the superpixels are used for matching operations in the best three algorithms.

All of these experiments show that as the number of pixels is much more than the number of superpixels, the covariance matrix obtained from pixels can model the variation in the data more accurately. On the other hand, using superpixel representatives for matching gives better results due to the suppression of noise effects and the simple representation of superpixel regions with their spectral means. Although the best combinations for all the compared methods give similar precision recall curves, the HUD algorithm is one level better as it gives the most consistent results considering the average precision values.

CHAPTER 7

SUMMARY AND CONCLUSIONS

In this thesis, firstly a detailed comparison between the adaptations of two superpixel extraction algorithms, the SLIC algorithm and boundary update based method, to hyperspectral images is performed in terms of boundary recall and undersegmentation error. The experiments reveal that the boundary update based superpixel extraction method is superior to the SLIC algorithm along with the advantages in memory efficiency and implementation complexity. Therefore, boundary update based superpixel extraction method is selected for superpixel based target detection. The superpixel representatives rather than pixels are utilized in the target detection algorithms for background modeling, matching operation and abundance estimation. Different combinations for each algorithm are considered. These combinations are compared in terms of precision and recall values, which evaluate the detection performance of the algorithms. As the last step, the best combinations of the target detection algorithms are compared by using the publicly available and experimentally prepared datasets.

The experiments on the superpixel based target detection methods first reveal that the usage of superpixels for the target detection algorithms gives better results than the pixels. Another important conclusion is that the detection performance of the hybrid detectors is superior to other algorithms. On the other hand, the ACE algorithm and its superpixel adapted versions indicate the good detection performances in addition to its simple implementation. One of the impressive results obtained from the experiments is that modeling the background by using pixels and employing superpixels for matching operations give the best performance for the superpixel based target detection algorithms. Finally, the obtained precision-recall curves show that choosing mean value as the representative of the superpixel region is the best option compared to the medoid and centroid values. With the performed experiments, the high potential of using superpixels instead of pixels for hyperspectral target detection applications is verified for the future.

REFERENCES

[1] "Hyperspectral Remote Sensing Scenes Dataset," `http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes`, Accessed: 2018-03-30.

[2] P. Neubert and P. Protzel, "Superpixel Benchmark and Comparison," in *Proceedings Forum Bildverarbeitung*. Karlsruher Institut für Technologie (KIT), 2012, pp. 205–218.

[3] "Precision-Recall Metrics," `https://upload.wikimedia.org/wikipedia/commons/2/26/Precisionrecall.svg`, Accessed: 2018-09-04.

[4] D. Manolakis, E. Truslow, M. Pieper, T. Cooley, and M. Brueggeman, "Detection Algorithms in Hyperspectral Imaging Systems," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 24–33, January 2014.

[5] J.M. Bioucas-Dias, A. Plaza, M. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral Unmixing Overview: Geometrical, Statistical, and Sparse Regression-Based Approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 354–379, April 2012.

[6] C.-I Chang, *Hyperspectral Data Processing: Algorithm Design and Analysis*, Wiley, Hoboken, New Jersey, 2013.

[7] J.R. Schott, *Remote Sensing: The Image Chain Approach*, Oxford University Press, Oxford, U.K., 1997.

[8] N.M. Nasrabadi, "Hyperspectral Target Detection," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 34–44, January 2014.

[9] I.S. Reed and X. Yu, "Adaptive Multiple-Band CFAR Detection of an Optical Pattern with Unknown Spectral Distribution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 10, pp. 1760–1770, October 1990.

[10] N. Dobigeon, Y. Altmann, N. Brun, and S. Moussaoui, "Linear and Nonlinear Unmixing in Hyperspectral Imaging," in *Data Handling in Science and Technology*, vol. 30, chapter 6, pp. 185–224. Elsevier, Amsterdam, Netherlands, 2016.

[11] X. Jin, S. Paswaters, and H. Cline, "A Comparative Study of Target Detection Algorithms for Hyperspectral Imagery," *Proc. of SPIE*, vol. 7334, April 2009.

[12] J. Broadwater and R. Chellappa, "Hybrid Detectors for Subpixel Targets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 11, pp. 1891–1903, November 2007.

[13] G. Camps-Valls, D. Tuia, L. Bruzzone, and J.A. Benediktsson, "Advances in Hyperspectral Image Classification: Earth Monitoring with Statistical Learning Methods," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 45–54, January 2014.

[14] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2281, November 2012.

[15] K.G. İnce, C. Çığla, and A.A. Alatan, "LASP: Local Adaptive Super-Pixels," in *International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 4092–4096.

[16] H.E. Taşlı, C. Çığla, and A.A. Alatan, "Convexity Constrained Efficient Superpixel and Supervoxel Extraction," *Elsevier Signal Processing: Image Communication*, vol. 33, pp. 71–85, April 2015.

[17] A. Psalta, V. Karathanassi, and P. Kolokoussis, "Modified Versions of SLIC Algorithm for Generating Superpixels in Hyperspectral Images," in *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*. IEEE, Aug 2016, pp. 1–5.

[18] A. Çalışkan, A. Koz, and A.A. Alatan, "Hyperspectral Superpixel Extraction Using Boundary Updates based on Optimal Spectral Similarity Metric," in *Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2015, pp. 1020–1023.

[19] "NASA Jet Propulsion Laboratory AVIRIS Dataset," `https://aviris.jpl.nasa.gov/alt_locator/`, Accessed: 2018-03-30.

[20] "Purdue University MultiSpec Dataset," `https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html`, Accessed: 2018-03-30.

[21] D. Manolakis, D. Marden, and G.A. Shaw, "Hyperspectral Image Processing for Automatic Target Detection Applications," *Lincoln Laboratory Journal*, vol. 14, no. 1, pp. 79–116, January 2003.

[22] D. Manolakis, R. Lockwood, T. Cooley, and J. Jacobson, "Is There a Best Hyperspectral Detection Algorithm?," *Proc. of SPIE*, vol. 7334, April 2009.

[23] M.T. Eismann, “Hyperspectral Target Detection,” in *Hyperspectral Remote Sensing*, chapter 14, pp. 646–716. SPIE Press, Bellingham, Washington, 2012.

[24] S.M. Kay, “Maximum Likelihood Estimation,” in *Fundamentals of Statistical Signal Processing: Estimation Theory*, chapter 7, pp. 157–219. Prentice Hall, Upper Saddle River, New Jersey, 1993.

[25] C.-I Chang, J. Liu, B. Chieu, H. Ren, C. Wang, C. Lo, P. Chung, C. Yang, and D. Ma, “Generalized Constrained Energy Minimization Approach to Subpixel Target Detection for Multispectral Imagery,” *Optical Engineering*, vol. 39, no. 5, pp. 1275–1281, May 2000.

[26] O.L. Frost, “An Algorithm for Linearly Constrained Adaptive Array Processing,” *Proceedings of the IEEE*, vol. 60, no. 8, pp. 926–935, August 1972.

[27] E. Truslow, “Performance Evaluation of the Adaptive Cosine Estimator Detector for Hyperspectral Imaging Applications,” M.S. thesis, Northeastern University, August 2012.

[28] H. Ren and C.-I Chang, “Target-Constrained Interference-Minimized Approach to Subpixel Target Detection for Hyperspectral Imagery,” *Optical Engineering*, vol. 39, no. 12, pp. 3138–3145, December 2000.

[29] H. Ren and C.-I Chang, “Automatic Spectral Target Recognition in Hyperspectral Imagery,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1232–1249, October 2003.

[30] D. Heinz, C.-I Chang, and M.L.G. Althouse, “Fully Constrained Least-Squares Based Linear Unmixing,” in *Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 1999, pp. 1401–1403.

[31] D. Heinz and C.-I Chang, “Fully Constrained Least Squares Linear Spectral Mixture Analysis Method for Material Quantification in Hyperspectral Imagery,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 3, pp. 529–545, March 2001.

[32] R. Bro and S. De Jong, “A Fast Non-Negativity-Constrained Least Squares Algorithm,” *Journal of Chemometrics*, vol. 11, pp. 393–401, February 1997.

[33] C.-I Chang, “An Information-Theoretic Approach to Spectral Variability, Similarity, and Discrimination for Hyperspectral Image Analysis,” *IEEE Transactions on Information Theory*, vol. 46, no. 5, pp. 1927–1932, August 2000.

[34] A. Çalışkan, E. Batı, A. Koz, and A.A. Alatan, “Superpixel Based Hyperspectral Target Detection,” in *Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2016, pp. 7010–7013.

[35] X. Jin and J. Han, “K-medoids clustering,” in *Encyclopedia of Machine Learning and Data Mining*, C. Sammut and G. I. Webb, Eds., pp. 697–700. Springer US, Boston, MA, April 2017.

[36] “Mikro-Tasarım Next Generation Imaging Sensors Product Catalog,” `http://www.mikro-tasarim.com.tr/Mikro-Tasarim_Brochure.pdf`, Accessed: 2018-07-09.

[37] “Liquid Crystal Tunable Filters Datasheet,” `https://www.perkinelmer.com/CMSResources/Images/44-140156DTS_010053A_01_VariSpec_DTS.pdf`, Accessed: 2018-07-09.

[38] “ASD FieldSpec 4 Hi-Res: High Resolution Spectroradiometer,” `https://www.malvernpanalytical.com/en/products/product-range/asd-range/fieldspec-range/fieldspec4-hi-res-high-resolution-spectroradiometer/index.html`, Accessed: 2018-07-10.

[39] “RIT Target Detection Blind Test,” `http://dirsapps.cis.rit.edu/blindtest/description/`, Accessed: 2018-07-12.

[40] S.E. Yüksel and A. Karakaya, “Fusion of Target Detection Algorithms in Hyperspectral Images,” *International Journal of Intelligent Systems and Applications in Engineering*, vol. 4, no. 4, pp. 103–110, October 2016.

[41] A. Karakaya and S.E. Yüksel, “Target Detection in Hyperspectral Images,” in *Signal Processing and Communications Applications Conference (SIU)*. IEEE, May 2016, pp. 1501–1504.

[42] D.W. Sun, “Spectral Preprocessing and Calibration Techniques,” in *Hyperspectral Imaging for Food Quality Analysis and Control*, chapter 2, pp. 61–62. Elsevier, Amsterdam, Netherlands, 2010.

[43] A. Levinshtein, A. Stere, K. Kutulakos, D. Fleet, S. Dickinson, and K. Siddiqi, “Turbopixels: Fast Superpixels Using Geometric Flows,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2290–2297, December 2009.

[44] D.M.W. Powers, “Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation,” *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, February 2011.

[45] “The Berkeley Segmentation Dataset and Benchmark,” `https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/`, Accessed: 2018-08-13.

[46] D. Snyder, J. Kerekes, I. Fairweather, R. Crabtree, J. Shive, and S. Hager, “Development of a Web-based Application to Evaluate Target Finding Algorithms,”

in *Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2008, vol. 2, pp. 915–918.