Democratic and Popular Republic of Algeria
Ministry of Higher Education and Scientific Research
University May 8, 1945 – Guelma
Faculty of Science and Technology
Department of Electronics and Telecommunications

Final Dissertation
For the Academic Master's degree
Domain: **Science and Technology**
Sector**: Electronics**
Speciality: **Instrumentation**

Oil Spill Detection in Hyperspectral Image Using Isolation Forest and SVM

Presented by:

Mekhancha Abderraouf

Boudefel Nassim

Under the supervision of:

Dr. BOUKAACHE Abdelnour

JUIN 2025

Acknowledgements

We would like to express our profonde gratitude to **Pr. Boukaache Abdennour,** Professor in the Department of Electronics and Telecommunications and our supervisor for guiding us through this research. His experience and scientific assistance were essential. His availability and the attention he paid to this work were a valuable asset in monitoring and advancing our thesis.

We also would like to thank all the jury's members that have accepted to examine our work.

We also would like to thank our loved ones, our families and friends who supported us throughout our university studies,

And express our gratitude to all the teachers who helped teach and guide us during our years at the university of 08 May 1945 of Guelma, and especially at the Department of Electronics and Telecommunications and its officials for their efforts, knowledge and patience.

Abstract

Oil spill detection has garnered increasing research interest in recent years due to the profound impact such incidents have on marine environments, natural resources, and the livelihoods of coastal communities. Hyperspectral remote sensing imagery offers a wealth of spectral information, which is highly advantageous for monitoring oil spills in complex oceanic scenarios. However, most existing methods rely on supervised or semi-supervised frameworks, requiring substantial effort to annotate a sufficient number of high-quality training samples. This process can be labor-intensive and time-consuming.

In this study, we use a novel approach which consists of an unsupervised oil spill detection method based on the isolation forest algorithm tailored for hyperspectral images (HSIs). The methodology begins with an estimation of noise variance across different spectral bands because noise levels can vary significantly. Bands severely affected by noise are subsequently discarded to improve data quality. Next, Principal Component Analysis (PCA) is employed to reduce the high dimensionality inherent in HSIs, facilitating more efficient processing.

The core of the approach involves estimating the probability that each pixel belongs to either the seawater or oil spill class using the isolation forest. This probabilistic information enables the automatic generation of pseudo-labeled samples through clustering algorithms, which serve as training data for subsequent classification steps. An initial detection map is then produced using support vector machines (SVM) on the dimension-reduced data.

To assess the effectiveness of our proposed method, we evaluated the method on dataset termed the Hyperspectral Oil Spill Dataset (HOSD), comprising eighteen hyperspectral images capturing oil spills over the Gulf of Mexico in 2010.

Résumé

La détection des marées noires suscite un intérêt croissant dans la recherche ces dernières années en raison de l'impact profond de tels incidents sur les milieux marins, les ressources naturelles et les moyens de subsistance des communautés côtières. L'imagerie hyperspectrale offre une richesse d'informations spectrales, particulièrement avantageuse pour surveiller les déversements d'hydrocarbures dans des scénarios océaniques complexes. Cependant, la plupart des méthodes existantes reposent sur des cadres supervisés ou semi-supervisés, nécessitant un effort considérable pour annoter un nombre suffisant d'échantillons d'entraînement de haute qualité. Ce processus peut être laborieux et chronophage.

Dans cette étude, nous adoptons une approche novatrice consistant en une méthode non supervisée de détection des marées noires basée sur l'algorithme Isolation Forest, adaptée aux images hyperspectrales (HSI). La méthodologie commence par une estimation de la variance du bruit à travers les différentes bandes spectrales, car les niveaux de bruit peuvent varier significativement. Les bandes sévèrement affectées par le bruit sont ensuite écartées afin d'améliorer la qualité des données. Puis, une analyse en composantes principales (ACP) est employée pour réduire la forte dimensionnalité inhérente aux HSI, facilitant un traitement plus efficace.

Le cœur de l'approche consiste à estimer, pour chaque pixel, la probabilité d'appartenance à la classe « eau de mer » ou « marée noire » à l'aide de l'Isolation Forest. Cette information probabiliste permet la génération automatique d'échantillons pseudo-étiquetés via des algorithmes de clustering, qui servent de données d'entraînement pour les étapes de classification ultérieures. Une carte de détection initiale est ensuite produite en utilisant des machines à vecteurs de support (SVM) sur les données à dimension réduite.

Pour évaluer l'efficacité de notre méthode proposée, nous avons évalués la méthode sur un ensemble de données hyperspectral complet, dénommé Hyperspectral Oil Spill Dataset (HOSD), comprenant dix-huit images hyperspectrales capturant des marées noires dans le golfe du Mexique en 2010.

ملخص

لقد حظيت تقنيات كشف انسكاب النفط باهتمام بحثي متزايد في السنوات الأخيرة نظراً للأثر العميق لمثل هذه الحوادث على البيئات البحرية والموارد الطبيعية وسبل معيشة المجتمعات الساحلية. توفر صور الاستشعار عن بُعد متعددة الأطياف فائقة الدقة ثروة من المعلومات الطيفية، مما يُعد ميزة كبيرة لمراقبة انسكابات النفط في سيناريوهات بحرية معقدة. ومع ذلك، تعتمد معظم الأساليب الحالية على أطر عمل خاضعة للإشراف أو شبه خاضعة للإشراف، مما يستلزم جهداً كبيراً لتوسيم عدد كافٍ من عينات التدريب عالية الجودة. وقد يكون هذا الإجراء شاقاً ويستغرق وقتاً طويلاً.

في هذه الدراسة، نعتمد على طريقة غير خاضعة للإشراف لكشف النفط المسكوب، تعتمد هذه الطريقة على خوارزمية (Isolation Forest) المصممة خصيصاً للصور متعددة الأطياف فائقة الدقة (HSIs). تبدأ المنهجية بتقدير تباين الضوضاء عبر مختلف النطاقات الطيفية، نظراً لاختلاف مستويات الضوضاء بشكل كبير. ثم تُستبعد النطاقات المتأثرة بشدة بالضوضاء لتحسين جودة البيانات. بعد ذلك، يُستخدم تحليل المكونات الرئيسية (PCA) لتقليل البُعدية العالية في هذه الصور، مما يسهل معالجة أكثر كفاءة.

يكمن الجوهر المتبع في تقدير احتمال ان يتضمن كل بكسل على «مياه البحر» أو «الانسكاب النفطي» باستخدام خوارزمية (Isolation Forest). تُمكّن هذه المعلومات الاحتمالية من توليد عينات ذات وسم زائف (pseudo-labeled) بشكل تلقائي عبر خوارزميات التجميع (clustering)، والتي تُستخدم كبيانات تدريب للخطوات التصنيفية اللاحقة. ثم تُنتَج خريطة كشف أولية باستخدام آلات المتجهات الداعمة (SVM) على البيانات المُختزلة البُعد.

لتقييم فعالية الطريقة المقترحة، جمعنا مجموعة بيانات شاملة متعددة الأطياف فائقة الدقة، أطلقنا عليها اسم مجموعة بيانات انسكاب النفط متعددة الأطياف (HOSD)، وتضم ثمانية عشر صورة التقطت انسكابات نفط في خليج المكسيك عام 2010.

List of Acronyms

Abbreviation	Full Term
HSI	Hyperspectral Imaging
PCA	Principal Component Analysis
SVM	Support Vector Machine
AVIRIS	Airborne Visible/Infrared Imaging Spectrometer
HOSD	Hyperspectral Oil Spill Dataset
iForest	Isolation Forest
RGB	Red Green Blue (color channels)
ROC	Receiver Operating Characteristic
AUC	Area Under Curve
EM	Electromagnetic (radiation)
MSS	Multispectral Scanner (Landsat sensor)
AIS	Airborne Imaging Spectrometer
PFRS	Portable Field Reflectance Spectrometer
ICA	Independent Component Analysis
MNF	Maximum Noise Fraction
LDA	Linear Discriminant Analysis
NLP	Natural Language Processing
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative
TPR	True Positive Rate (Recall/Sensitivity)

Abbreviation	Full Term
FPR	False Positive Rate
DP	Detection Precision
RBF	Radial Basis Function (Kernel)

Table of Contents

List of figures

List of tables

Introduction

Introduction

Oil spill detection is a necessary and important task for us to tackle because of oil leaks high and bad effects on the environment due to the accidents caused during oil explorations and transportation happening around the world, which leads to severe pollution in the marine environment and damages coastal species.

If the oil leaked would not be monitored properly, the oil slick would find its way to the coast following the sea waves, which leads to high threats to coastal species from fish to coral reefs and even human health would be affected.

To detect and monitor the oil spills, we need to apply sensing techniques specially in remote and inaccessible areas on a large scale, it is also possible to predict the speed and direction of the oil movements using multi-temporal data and drift predictions models, which play an important role in facilitating clean-up tasks.

Over the past decades, remote sensing has been extensively explored for oil spill detection and monitoring.

Early methods utilizing airborne visible (VIS) and infrared (IR) data faced limitations such as poor separability between oil spills and surrounding objects. In contrast, active microwave sensors, particularly synthetic aperture radar (SAR), have become prominent due to their ability to operate in all weather conditions and during day and night. SAR detects oil slicks as dark spots caused by the inhibitory effect of oil on capillary waves, reducing backscatter.

Nonetheless, challenges remain in distinguishing actual oil spills from other phenomena such as grease ice or internal waves that also produce dark patches in SAR imagery. Additionally, SAR data suffer from high costs, low revisit frequencies, and limited swath widths, prompting interest in multi-platform SAR and supplementary optical sensors like multispectral and hyperspectral imagers.

Multispectral sensors, such as MODIS and Landsat, have also been employed for oil spill detection, relying on spectral and spatial information. However, their relatively coarse

spatial resolutions limit effectiveness for identifying small spills. Conversely, hyperspectral sensors mounted on aircraft offer both rich spectral and high spatial resolution, enabling detailed analysis of oil spill features and emission types like water-in-oil (W/O) and oil-in-water (O/W). These sensors facilitate advanced machine learning techniques, such as spectral shape matching and feature extraction, which improve detection accuracy and reduce false alarms. Overall, hyperspectral data represent a promising avenue for precise oil spill identification, especially when combined with sophisticated analytical methods.

Even though we still face several challenges to detect the oil spill accurately using hyperspectral images due to:

- The lack of dataset in regards of oil spills either there are none or you need to pay some money to acquire some.
- Training samples requires either a supervised or semi-supervised approach which is expensive and time consuming
- Bad image noise due to low lighting, shadow and bad weather results in corrupted hyperspectral images that leads to corrupted spectral bands and this has a negative effect on the accuracy of the oil spill detection.

In order to overcome these issues in this work we have used an unsupervised oil spill detection method based on isolation forest followed by other steps mentioned in chapter 2, Where we used a novel hyperspectral remote sensing database for oil spill detection, which is a publicly available benchmark dataset [1].

The manuscript is organized in three chapters. In the first one, we take a brief deviation and introduce an aspect of what is hyperspectral imaging and what are the tools used to capture these images, its origin and history, and differences from normal images.

Chapter 2 describes the different steps, algorithms and methods that are combined to produce the clustered image and detect oil spills. Finally, the last chapter contains the obtained results and discussion of numerical values and the overall evaluation of the method. Finally, a conclusion is given.

Chapter 1: Hyperspectral images (HSI)

1.1 Introduction

Hyperspectral imaging is all about the fundamental principles of how light interacts with materials and how this interaction enables precise material identification, it begins with electromagnetic radiation emitted by the Sun, where different wavelengths are absorbed or reflected by objects on Earth, forming the basis of spectral analysis.

Each material has a spectral signature, a unique pattern of reflectance highlighting their significance in remote sensing applications.

In this chapter, we explore the technological aspects of hyperspectral imaging, including the functioning of hyperspectral cameras and spectrometers, and distinguishing between various acquisition techniques such as point scanning and line scanning. Additionally, the historical development of HSI is traced from its origins in spaceborne Earth observation missions to its current widespread applications in environmental monitoring, geology, agriculture, and beyond.

Furthermore, we need to emphasize the importance of hyperspectral sensors, like AVIRIS, demonstrating their capabilities for detailed surface characterization and their role in critical environmental applications such as oil spill detection. We also underscore the versatility of HSI across multiple fields, illustrating its expanding role in industry, medicine, homeland security, and resource management.

1.2 Hyperspectral Imaging (HIS)

HIS is the study of light interaction with materials and how it helps identifying them, so let us take a quick dive on how sunlight and electromagnetic energy works.

Electrical magnetic energy (EM) radiates from the sun in waves, these waves are varied in size, most of these waves are invisible to our human eyes (figure 1.1) but the amazing thing is that when these waves reach the surface of the earth, they either get absorbed or reflected by objects, we call it (Spectrum). This phenomenon depends on their structures where each object has its own spectral signature (Spectroscopy) and with this information we can measure the intensity of the absorption or reflection using a special camera (figure 1.2) through a process known as hyperspectral imaging.

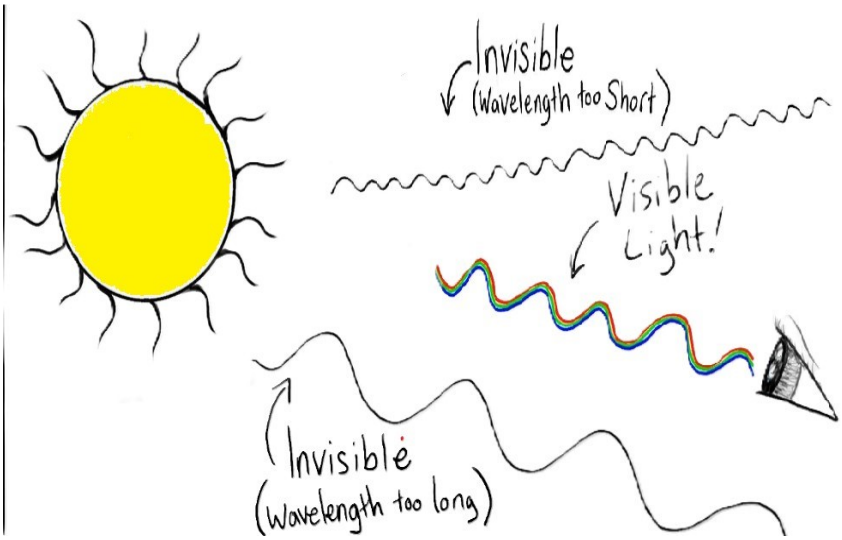


Figure 1.1: visible and invisible wavelengths [2].

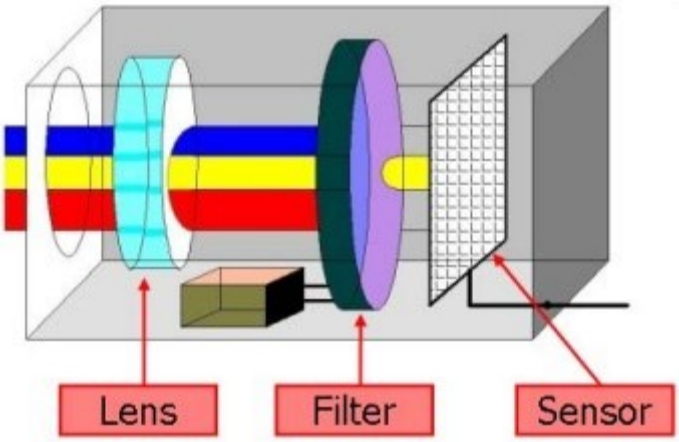


Figure 1.2: Hyperspectral Camera scheme [3].

Each material has a unique spectral signature which allows the light to behave in a certain way depending on that signature.

The spectrum is the amount of light in different wave lengths which shows how much light is emitted, reflected or transmitted from the material, in other words spectrum shows how much a certain color is contained in the light and for that aspect we can use a spectral

signature to identify our materials because we concluded that each material has its own signature similar to the human's finger print.

Where a normal digital camera captures normal visible light waves reflected off of objects and recorded the information in just 3 bands (RED, GREEN, BLUE) similar to a human eye (figure 1.3), it can't capture the electromagnetic waves which contain thousands of wave lengths ranging from large radio waves to very small Gamma waves. In conclusion, the (EM) contains thousands of waves that are transformed into bands and to capture them we need to use a hyperspectral camera or a spectrometer.

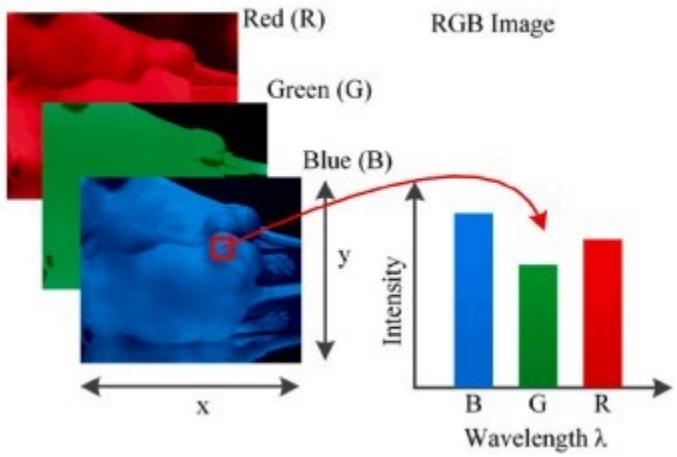


Figure 1.3: RGB image [4].

1.3 Hyperspectral camera (Spectrometer)

An instrument that splits the incoming light (reflected light) into its individual wavelengths or spectral bands, it provides a two-dimensional image of a scene while simultaneously recording the spectral information of each pixel in the image, this detailed spectral information allows for more precise material identification, chemical composition analysis, and environmental monitoring.

A hyperspectral image has two spatial dimensions (Sx and Sy) and one spectral dimension (Sλ) which forms a 3D hyperspectral data cube (figure 1.4).

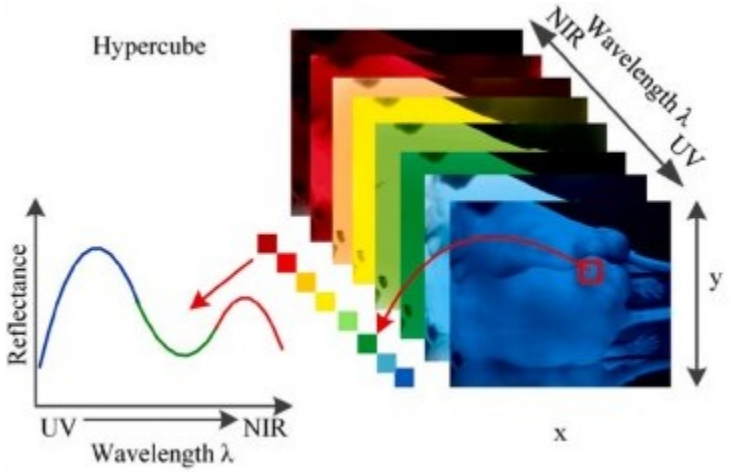


Figure 1.4: 3D hyperspectral data cube [4].

1.3.1 Spatial Resolution

It defines the clarity of the image and not the number of pixels in an image, its characteristics depends on the design of the sensors in terms of its field of view or altitude, take for example a patch of land that we want to take a picture off, the smaller the size of the patch the higher the details we can get.

1.3.2 Spectral Resolution

It is the number of spectral bands and range of electromagnetic spectrum measured by the sensor, where the resolution corresponds to the number of bands. The higher the number of bands, the better the resolution.

1.3.3 Temporal Resolution

Usually defined in days and it is the time needed by the sensor to revisit and obtain data from the exact same location, which means the higher the revisit frequency the higher the temporal resolution is.

1.3.4 Understanding Spectral Signatures

Hyperspectral sensors allow us to measure all types of electromagnetic energy within a specific range as it interacts with materials (absorb, transmit and reflect).

Reflectance is all about measuring the electromagnetic energy bouncing back from a material's surface; it can range from [0-100], where 0 means that the material absorbed the entire light and 100 means that all the light was reflected.

To be specific the reflectance values of different materials on the surface of the earth such as soil, forest, water and minerals can be plotted into spectral signatures (spectral response curves) and compared (Figure 1.5).

The more spectral resolution of an imaging sensor, the more classification information can be extracted from spectral signatures.

Hyperspectral imagery has been utilized by geologists for mapping the land and water resources as well as to map heavy metals and other hazardous wastes in active mining areas [6].

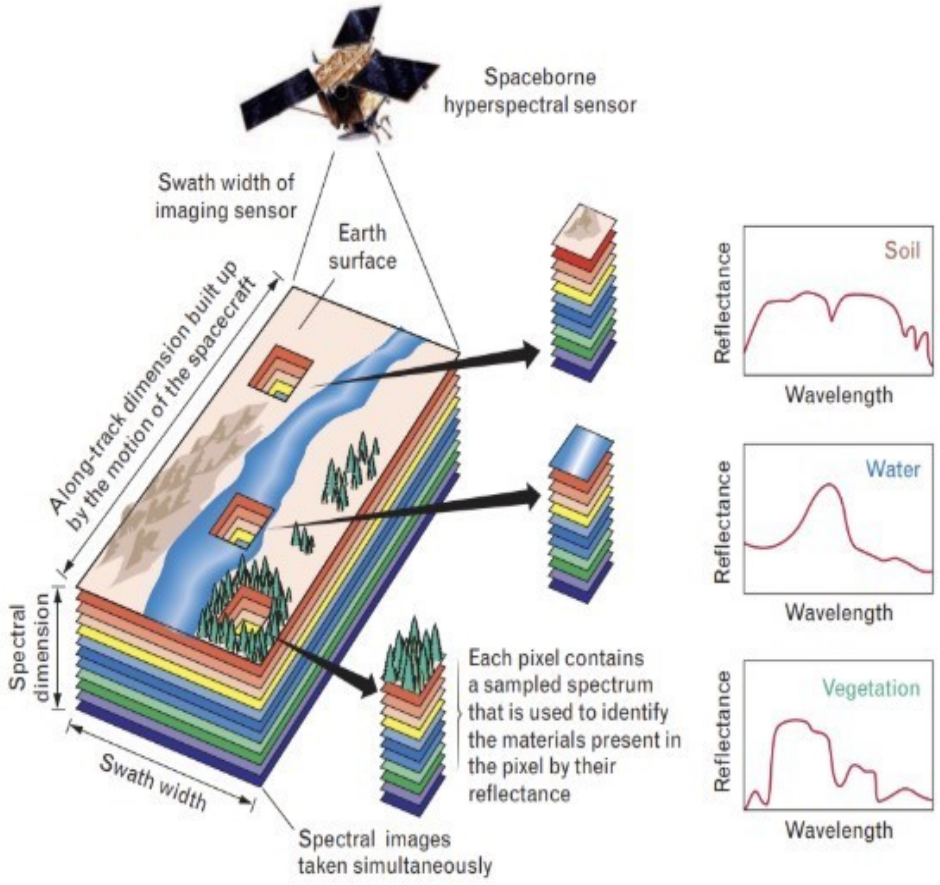


Figure 1.5: Spectral signature of different materials [5].

1.4 The Origin of Hyperspectral Imaging

Hyperspectral imaging (HSI) is a powerful and non-destructive analytical technique that captures both spatial and spectral data across a wide range of narrow and contiguous wavelengths. Originally developed for Earth observation in remote sensing, it has since evolved into a versatile tool applied in numerous fields such as agriculture, food quality control, environmental monitoring, and biomedical diagnostics. By collecting a complete spectral signature for each pixel in an image, HSI enables detailed identification of surface and material properties, far beyond the capabilities of conventional imaging systems.

The conceptual roots of hyperspectral imaging lie in spectral remote sensing, a domain that began with the launch of Landsat 1 in 1972, then known as the Earth Resources Technology

Satellite (ERTS-1). This satellite carried the Multispectral Scanner (MSS), enabling the capture of Earth surface data in several discrete bands. Though revolutionary at the time, multispectral data lacked the spectral resolution needed for more precise material discrimination [6].

During this period, researchers at the NASA Jet Propulsion Laboratory (JPL), including Goetz and the late Gene Shoemaker, began analyzing MSS data for geological mapping, particularly on the Coconino Plateau in Arizona. They encountered challenges interpreting subtle color variations in the imagery, which could not be adequately explained without direct spectral measurements from ground samples. This need led to the development of the first portable field reflectance spectrometer (PFRS) in 1974, capable of capturing reflectance data across the 0.4 to 2.5 μm range the full solar-reflected spectrum [7]. These early efforts were instrumental in influencing the design of future remote sensing instruments, such as the addition of Band 7 to the Landsat Thematic Mapper.

Recognizing the limitations of multispectral imaging, Goetz and colleagues at JPL proposed the concept of imaging spectrometry, formally defined as the acquisition of images in hundreds of contiguous spectral bands such that a full radiance spectrum could be obtained for each pixel [8]. In a 1985 Science publication, this vision culminated in the development of the Airborne Imaging Spectrometer (AIS), the first sensor of its kind, and it was in this context that the term “hyperspectral imaging” was first coined by Jerry Solomon [8].

A major step forward came with the creation of AVIRIS (Airborne Visible/Infrared Imaging Spectrometer), developed by NASA JPL in 1987 as a more advanced successor to the AIS. AVIRIS was designed to acquire high-quality hyperspectral data across 224 contiguous spectral bands, covering the visible to shortwave infrared range (400–2500 nm). Its primary goal was to measure and characterize Earth's surface and atmosphere with greater precision, allowing for the identification and mapping of surface materials, vegetation types, mineral deposits, and atmospheric constituents [9].

Since its inception, AVIRIS has been flown over a wide variety of geographic regions including deserts, forests, agricultural areas, and volcanic zones providing critical data for environmental monitoring, land use studies, and climate research [9].

Despite the technical limitations of the time, such as limited onboard processing power and dependence on centralized computing facilities, AVIRIS established a new standard for airborne hyperspectral sensing. It confirmed that high-resolution, laboratory-quality spectral measurements could be achieved remotely and effectively over broad areas. The success of AVIRIS greatly influenced both research and instrument development in the decades that followed [9].

As sensor technology and data processing capabilities improved, hyperspectral imaging found applications beyond its original scope. By the late 1990s, researchers such as Lu and Chen demonstrated the technique's potential in agriculture and postharvest quality evaluation, notably in detecting defects in fruits like apples [10]. Soon, the food industry began to adopt HSI as a non-invasive means to evaluate internal attributes such as sugar content, acidity, and texture properties invisible to the human eye or standard RGB imaging systems.

Researchers including Gowen et al. and Nicolaï et al. expanded the field further, using hyperspectral imaging for chemical imaging, allowing for spatial visualization of biochemical content [11-12]. These developments marked a shift toward real-time industrial applications, where quality and safety control could be automated, non-invasive, and highly accurate.

Today, hyperspectral imaging continues to evolve rapidly, driven by advances in optics, electronics, data science, and machine learning. From its origins in space-based Earth observation missions to its growing presence in precision agriculture, food inspection, medical diagnostics, and environmental science, HSI has proven to be a transformative technology. It serves as a prime example of how foundational innovations in one field such as NASA's efforts in planetary and Earth sciences can ripple across disciplines, solving new challenges and inspiring new research directions.

Table 1.1: The current space and airborne satellite hyperspectral sensors [4].

	Sensor	*Origin*	*Spectral Range*	*No. of spectral bands*	*Spectral Resolution (nm)*	*Operational Altitude (Km)*	*Spatial Resolution (m)*
Satellite Based	Hyperion	NASA, UK	352-2576	220	10	707 (7.7 km)	30
	PROBA-CHRIS	ESA, UK	415-1050	19 63	34 17	830 (14 km)	1736
Airplane Based	AVIRIS	Jet Propulsion Laboratory, USA	400-2050	224	10	-	-
	CASI	Itres, Canada	380-1050	288	<3.5	1-20	1-20
	AISA HyMap	Specim, Finland Integrated Spectronics, Australia	400-970 440-2500	244 128	3.3 15	1-20 -	1-20 -
UAV Based	Head Well Hyperspec	Headwall Photonics, USA	400-1000	270 Nano 324 Micro	6 Nano 2.5 Micro	<0.15	0.01-0.5
	UHD 185 Firefly	Cubert, Germany	450-950	138	4	<0.15	0.01-0.5

1.5 Different usages of hyperspectral imaging

This is a rapidly growing field and has a great variety of applications such as: military, industrial, commercial (food safety and quality), medical fields, water food and resources management, agriculture, forensics, homeland and defense security, plant detection and fire prediction, weed and crop discrimination [4], and most importantly our main theme for this thesis which is oil spill detection.

1.6 Conclusion

In this chapter, we have explored the fundamental principles and technological advancements of hyperspectral imaging (HSI). Starting from the interaction of sunlight and electromagnetic radiation with Earth's surfaces, we examined how spectral signatures enable the precise identification of materials. The chapter outlined the operational mechanisms of hyperspectral sensors, and traced the historical evolution of HSI from its origins in spaceborne remote sensing to its diverse modern applications. Emphasizing the sensor's high spectral and spatial resolution capabilities, we highlighted its significant role in environmental monitoring, including critical tasks such as oil spill detection. Overall, this chapter provides a comprehensive understanding of hyperspectral imaging technology,

laying the groundwork for its application in the subsequent analysis and detection methods discussed in this thesis.

Chapter 2: HSI clustering

2.1 Introduction

This Chapter provides a comprehensive, step-by-step methodological explanation of oil spill detection pipeline for hyperspectral data, which integrates Principal Component Analysis (PCA), Isolation Forests, and Support Vector Machines (SVM). Data preprocessing, dimensionality reduction, unsupervised anomaly scoring, supervised refinement, and performance evaluation is discussed in detail.

2.2 Machine Learning

Machine Learning is a way where computers learn from Data and use what was learned to make judgments. It is divided into many techniques, from basic linear regression to advanced deep learning models.

Machine learning is also known as a branch of artificial intelligence which instructs computers to analyze data and extract conclusions by improving their performance on a specific task through data analysis instead of a specific instruction or programming.

It is used by scientists for diverse purposes, and is a subset of artificial intelligence, where (AI) can do problem-solving, decision-making and spotting patterns and gaining knowledge..., (AI) aims to mimic human intellect and require human-level cognition and reasoning.

(ML) focuses on specific tasks such as image recognition, recommendation systems, natural language processing (NLP), healthcare, financial services, language translation and more, which all depend on these core components (figure 2.1). It focuses on crafting algorithms, models that enable computers to learn from data enhancing performance progressively, where we can consider machine learning as one of the many tools used by (AI).

- Data: can be texts, numbers, images or any information that can be processed by a computer, not to forget the quality and quantity of this said data which plays a significant role in (ML).

- Algorithms: are the heart of the process, and are responsible for learning patterns and relationships in our Data, where it can be supervised or unsupervised or even reinforcement learning.
- Training: used in supervised learning, it allows the algorithm to learn from the data where the correct answer is and make correct predictions when new data is presented.
- Model: it uses the algorithms to encapsulate the patterns and relationships it has learned from the data during training, after that it can be used for making predictions or decisions.
- Testing and validation: it is essential after training, to ensure the model's ability to apply its new found knowledge on unseen examples.
- Deployment: after training and validation, we can be integrated into real-world via software, systems or devices to automate tasks, assist in decision making or make predictions.

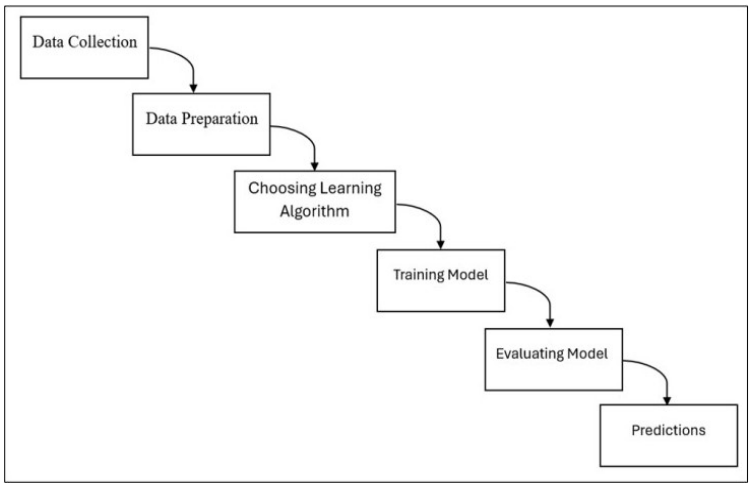


Figure 2.1: Machine learning [14].

The rapid advancement of machine learning has brought about significant developments in supervised and unsupervised learning which they are two fundamentals used in a wide range of applications however, these learning methods are often faced by various challenges from complexities of data labeling and overfitting, limiting its scalability and

generalization in supervised learning to the intricacies of clustering and noise management and interpretability in unsupervised learning.

2.2.1 Supervised Learning

Trained algorithms using labeled data, meaning the input data comes with corresponding correct outputs, where the goal is enabling the algorithm to classify new data or make predictions based on what patterns were learned during training (Figure 2.2).

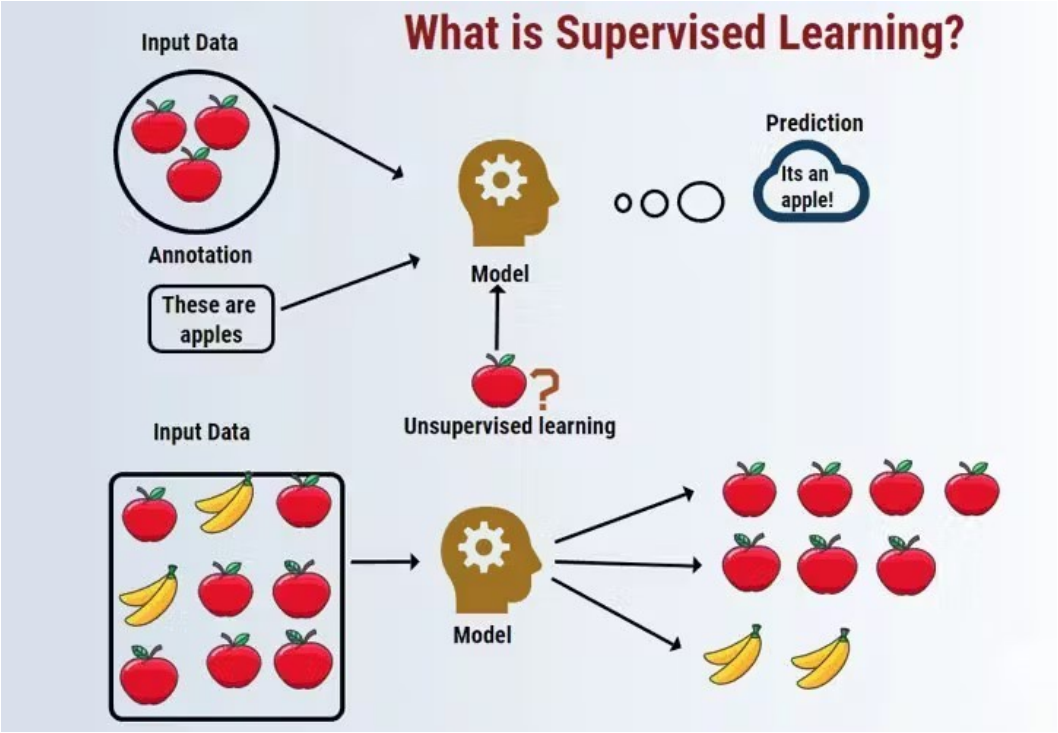


Figure 2.2: Supervised Learning Diagram [14].

Supervised learning plays a central role especially when the objective is predicting or classifying based on labeled data.

2.2.2 Supervised Learning Algorithms

- **Linear Regression:** is a foundational algorithm in supervised machine learning and is commonly used to forecast future outcomes by establishing the relationship between a dependent variable (target) and one or more independent variables (features) where the objective is to predict a continuous numeric output. In the case

where there is only one independent variable and one response variable, the model is called "simple linear regression". In contrast, "multiple linear regression" is used when multiple independent variables are involved.

- **Logistic Regression:** where Linear Regression is better used and suited for categorical dependent variables with binary output like ‘’true’’ and ‘’false’’ or ‘’yes’’ and ‘’no’’, we use logistic regression to tackle binary qualification issues (spam identification), while both graphical and non-graphical seek to discover correlations between data inputs.
- **Decision Trees:** widely used in supervised machine learning for tasks such as classification and regression, where the algorithm works by splitting the data into smaller subsets based on the values of input features. Each internal node in the tree is formed by a decision based on a specific feature, branches illustrate the possible outcomes of those decisions, and leaf nodes provide the final prediction or classification.
- **Support Vector Machine (SVM):** created by Vladimir Vapnik, a well-known supervised learning model applied to both data classification and regression, but is used primarily in classification issues, it works by creating a hyperplane gapping a large space between two sets of data (for example oranges and apples) into distinct groups.
- **Naïve Bayes:** based on the Bayes theorem including Bernoulli Naïve Bayes, Multinominal Naïve Bayes and Gaussian Naïve Bayes. It is a classification approach founded on the assumption of conditional independence among classes. Commonly used in text classification, recommendation engines and spam identification.

2.2.3 Unsupervised Learning

Focuses on uncovering patterns, structures, or relations within labeled data. Its capability to identify similarities and distinctions in data positions is the perfect solution for tasks like exploratory data analysis, strategies for cross-selling customer segmentation, and image recognition (figure 2.3).

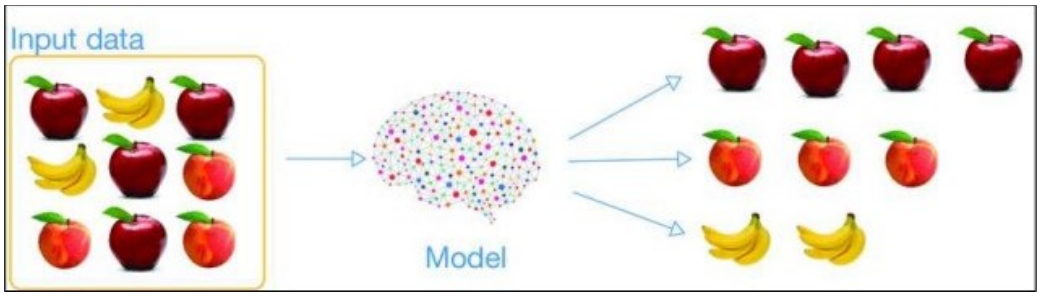


Figure 2.3: Unsupervised Learning Diagram [14].

2.2.4 Unsupervised Learning Algorithms

Clustering is a data organizing approach which works as follows: similar attributed items are grouped together within a cluster, and distinct items are assigned to separate clusters. In short it categorizes data objects into groups according to if they share or do not share characteristics (figure 2.4).

The most used unsupervised learning algorithms depicted are as follows:

- **K-Means:** groups data points into ‘’K’’ clusters, ‘’K’’ is a number you specify. The point is to assign each data point to the cluster with the closest center minimizing the distance between the data points and their center, the process is then repeated until the clusters are as tight and distinct as possible.
- **Hierarchical Clustering:** dividing data into hierarchy of clusters represented as a tree-like structure, every data point is designated as an individual distinct cluster. Assigning data to an existing cluster, or merging two clusters iteratively a novel cluster can be created.
- **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):** is a clustering algorithm distinguishing between high-density and low-density clusters, so it is a density-based clustering technique used for locating clusters of similar large datasets.

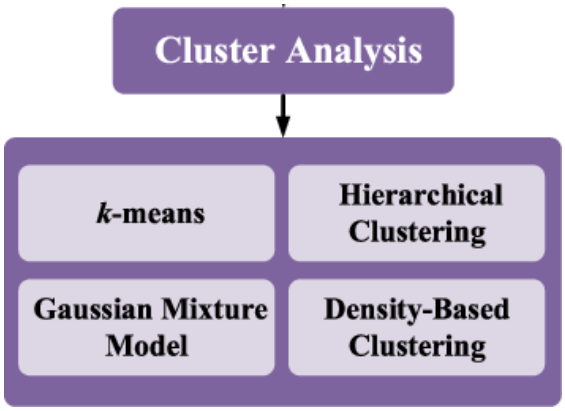


Figure 2.4: Unsupervised Learning algorithms.

2.2.5 Challenges encountered in supervised and unsupervised learning

- **Supervised Learning:**
  - Expensive learning algorithms requiring significant computational resources.
  - The need for labeled data, where obtaining and annotating a large dataset can be time-consuming and expensive.
  - Models in supervised learning tend to perform well on data similar to the training set but can struggle to generalize to new or unseen data.
  - Imbalanced datasets, where one class significantly outweighs the other, leading to biased models and difficult to make accurate predictions.

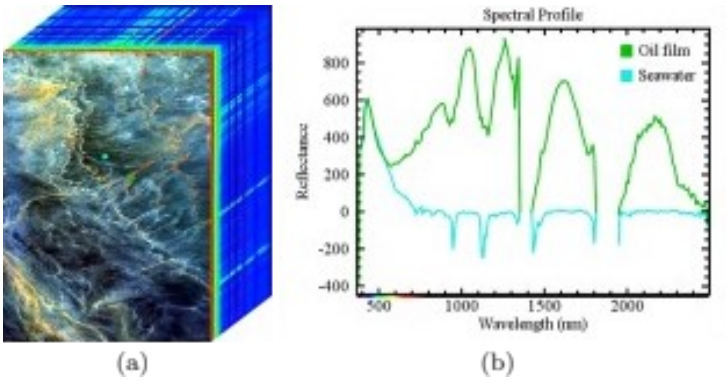
- **Unsupervised Learning:**
  - Influenced by the choice of algorithms and decision-making in performance.
  - Subjective clustering quality evaluation, and there might be no ground truth to compare against.
  - Impacting the quality of clustering results is subjugated to selecting the optimal number of clusters which is an unsolved problem.
  - The lack of clear interpretations leading to challenges in discovering a significance in patterns.

- Computationally expensive and poorly scalable to large datasets in some unsupervised learning algorithms such as hierarchical.
- Sensitivity to noisy data and outliers leading to incorrect results [14].

2.3. Oil Spill Detection Using HSI

2.3.1. Anomaly Detection

Anomaly detection in hyperspectral images focuses on identifying pixels whose spectra significantly deviate from the background (Figure 2.5).



(a) Hyperspectral oil spill image. (b) Spectral curve

Figure 2.5: The spectral curve of different objects [1].

Early approaches, such as the Reed-Xiaoli (RX) detector [15], model the background distribution using Gaussian statistics and compute a Mahalanobis distance for each pixel. Subsequent variants incorporate robust covariance estimation [16], subspace projections [17], and kernel-based variants [18]. The challenge remains that real-time applications require both accuracy and computational efficiency.

2.3.2. Dimensionality Reduction Using PCA

Hyperspectral data typically have hundreds of bands (e.g., AVIRIS collects 224 bands). Many bands are highly correlated; hence, dimensionality reduction is essential. Principal Component Analysis (PCA) [19] is widely used to project data into an orthogonal subspace of smaller dimensionality, capturing the majority of variance. Alternatives include Independent Component Analysis (ICA) [20], Maximum Noise Fraction (MNF) [21], and supervised linear discriminant analysis (LDA) [22]. PCA's strength lies in its simplicity and the ease of reconstructing approximate spectra.

Principal Component Analysis (PCA) is a widely used linear transformation technique for reducing the dimensionality of hyperspectral images (HSIs). Hyperspectral data typically consist of hundreds of contiguous spectral bands, many of which contain redundant or correlated information. This high dimensionality increases computational complexity and storage requirements while potentially degrading classification and detection performance due to the "curse of dimensionality." PCA addresses these challenges by transforming the original high-dimensional data into a lower-dimensional subspace while retaining the most significant spectral variations [36].

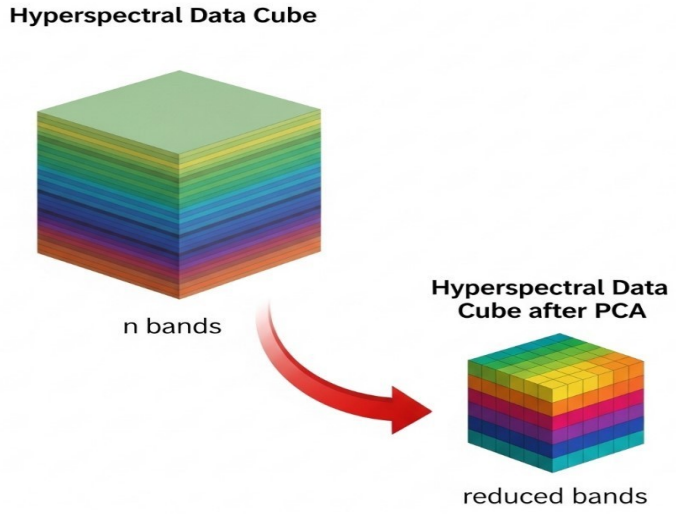


Figure 2.6: Reducing the dimensionality of data cube using PCA.

PCA operates by projecting the original spectral bands into a new orthogonal coordinate system defined by eigenvectors (principal components, or PCs) of the data covariance matrix. It reorients the data from its original band-based axes into a new set of axes (dimensions) that are orthogonal (perpendicular and uncorrelated). These new axes are the Principal Components (PCs), and this is useful because in the original data, bands may overlap in information (e.g., two bands might respond similarly to vegetation), PCA finds new directions (PCs) where the data varies the most, eliminating redundancy. The principal components are nothing but the new coordinates of points with respect to the new axes. The result of the projection will be represented in recombining the original spectral bands as weighted sums (projections) onto these new PC axes [37].

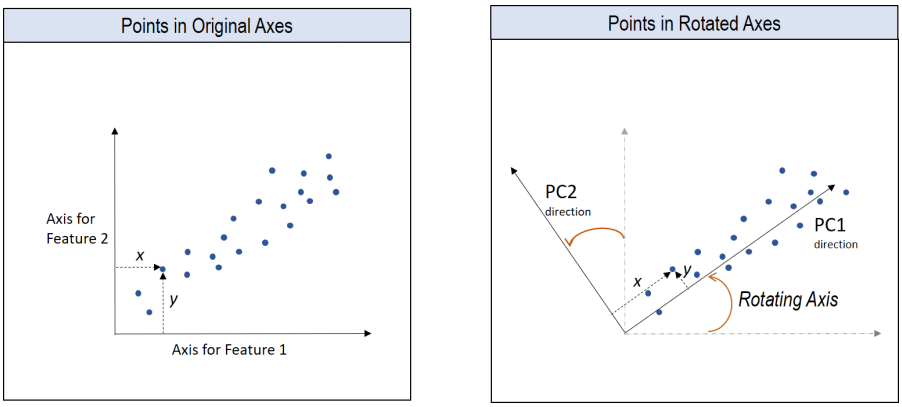


Figure 2.7: The rotation of original axes [45].

- ***PCA computation***

1- First, we begin by calculating the covariance matrix of the dataset, it helps us understand how variables (features) in the dataset relate to each other. For this reason, we center the data by removing the mean (Mean Subtraction) because PCA is sensitive to the scale of data.

We do it by computing the mean (average) of each feature (column) across all samples then subtracting this mean from every data point in that feature.

$$X_{centered} = X - \mu. \qquad 2.1$$

2- Now we compute the Covariance Matrix, it tells us how features vary together to calculate it, we multiply the transposed centered data ($X^T_{centered}$) by itself ($X_{centered}$), then scale by $\frac{1}{n-1}$ (for unbiased estimation in statistics), we represent it mathematically as:

$$C = \frac{1}{n-1} X^T_{centered} X_{Centered} \qquad 2.2$$

the result is a d × d symmetric matrix where (C_{ii}) is the variance of feature *i* [38].

3- The next step is finding the Eigenvectors of the Covariance Matrix. Eigenvectors define the directions of maximum variance (Principal Components), while eigenvalues tell us how much variance each PC captures, they tell us the directions where data varies the most, while eigenvalues quantify their importance.
We start by solving the eigenvalue equation:

$$Cv = \lambda v \qquad 2.3$$

where:

C = covariance matrix.

v = eigenvector (direction).

λ = eigenvalue (magnitude of variance).

The result will be a set of eigenvectors (v_1,v_2, ... ,v_d) and their corresponding eigenvalue (λ_1,λ_2,...,λ_d).

Then to rank Principal Components by importance (highest variance first) we sort Eigenvectors by Eigenvalues. We start by sorting the eigenvalues in descending order:

$\lambda_1>\lambda_2>...>\lambda_d$, then we reorder eigenvectors accordingly. This will give us:

v_1 = first Principal Component (direction of max variance).

v_2 = second Principal Component (next best direction, orthogonal to v_1), etc [39].

4- The last step is transforming the original data into the new PCA space, by selecting the top-k Eigenvectors. We chose the first *k* eigenvectors (where $k < d$) that capture,

e.g., 95% of total variance, forming a projection matrix W (size d × k) with these eigenvectors as columns.

Now, we transform the data by multiplying the centered data by the projection matrix:

$$X_{PCA} = X_{centered} \cdot W \qquad 2.4$$

The result X_{PCA} represents a new dataset with *k* dimensions instead of *d* [40].

After building the principal components of the dataset we simultaneously compute the weights of the principal components. The weights are the coefficients of the original variables (features) in each principal component (PC). They define how much each original feature contributes to a PC [41].

Since PCs are ranked by importance (variance), we can discard weaker ones, reduce dimensions while keep most information. And thus, orthogonality ensures that each PC is independent (*uncorrelated*), meaning that PC1 explains the most variance then PC2 explains the next most, without overlapping with PC1, etc. [42].

Despite its advantages, PCA may not always improve detection performance, particularly when the target of interest (e.g., an oil spill or mineral deposit) has a spectral signature similar to its background. Since PCA prioritizes high-variance features, subtle but critical spectral differences may be suppressed in the lower-variance components, reducing detectability. Alternative methods, such as Independent Component Analysis (ICA) or supervised feature extraction, may be more effective in such cases [43].

In summary, PCA is a powerful unsupervised tool for hyperspectral dimensionality reduction, offering computational efficiency and enhanced classification performance. However, its effectiveness depends on the data characteristics, and careful consideration is needed when applying it to specific detection tasks [45].

2.3.3. Isolation Forest for Unsupervised Anomaly Scoring

Isolation Forest (iForest) [23] is an ensemble-based anomaly detection algorithm that isolates observations in a data set. The key insight is that anomalies, being “few and different,” get isolated more quickly in a random partitioning tree structure check (figure

2.8).

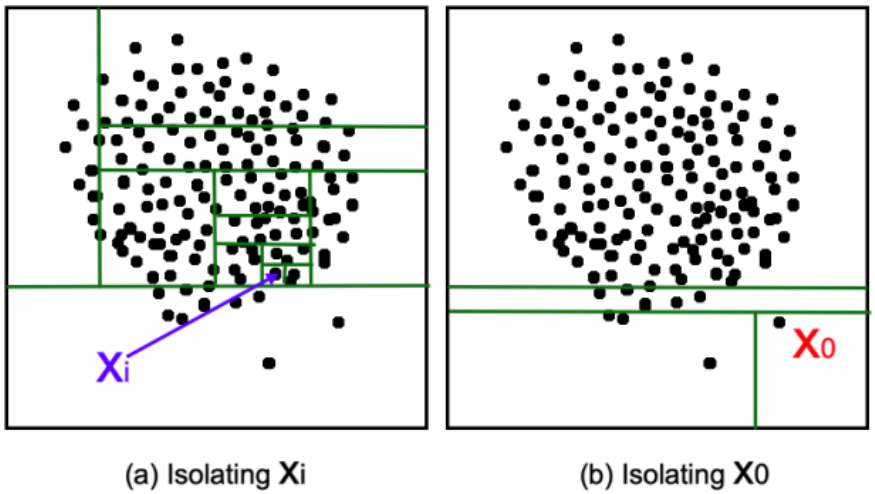


Figure 2.8: Anomalies (Xi) and (X0) are isolated faster than normal data [24].

Unlike distance-based or density-based methods (e.g., k-nearest neighbors [25], local outlier factor [26]), iForest directly models the notion of isolation and scales well to large datasets.

2.3.3.1. Concept of Isolation Forest

The term isolation means 'separating an instance from the rest of instances', and because anomalies are few and different which makes them easier to isolate by partitioning of instances repeatedly until all instances are isolated. To demonstrate the idea of anomalies are more susceptible to isolation under random partitioning, we illustrate an example in (figure 2.9) is partitioning of a normal point which requires more partitions to be isolated versus the anomaly in (Figure 2.6.b) that requires less partitions to be isolated, partitions are generated randomly. Since repeated partitioning can be represented by tree structure it means that the number of partitions required to isolate a point is equivalent to the path length from the root node to a terminating node [27].

Suppose a data-induced Binary Decision Tree with an anomaly present. The assumption" few and different" implies anomalies are decided closer to the root, and normal points are deeper in the tree. The binary tree is built to isolate all the points and measure their individual Path Lengths from the root.

- **Isolation Tree:** T is either an external-node with no child, or an internal-node with one test and exactly two daughter nodes (T_l, T_r) of an Isolation Tree. A test consists of an attribute q and a split value p such that the test q < p divides data points into T_l, and T_r.

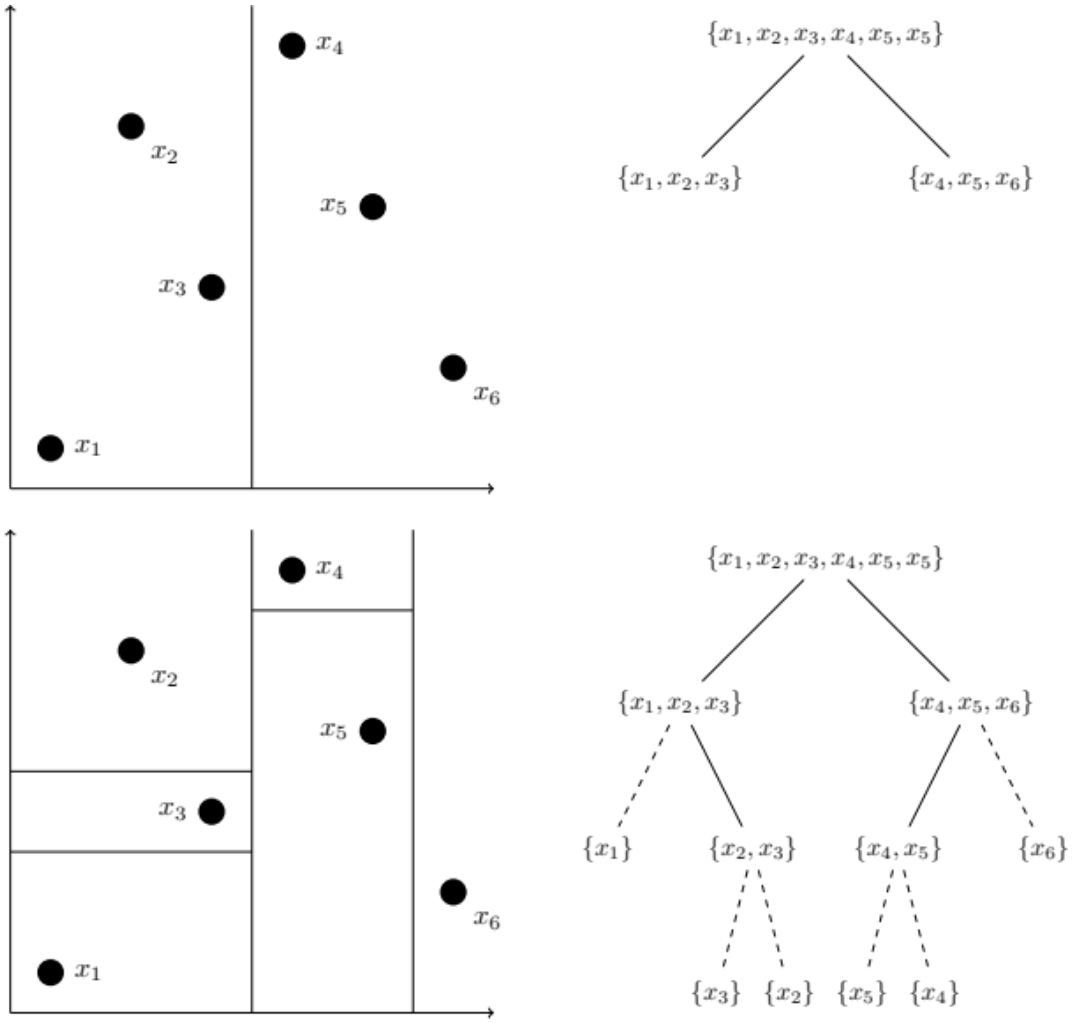


Figure 2.9: An Example of a Construction of an Isolation Tree [31].

- **Path Length:** h(x) of a point x is measured by the number of edges x traverses an iTree from the root node until the traversal is terminated at an external node.
- **iForest:** of size t is an ensemble of t iTrees. In short, DATA are matrices of real numbers of a dimension $N \times P$, where N is number of rows and P is number of features. In training stage, the Isolation Forest algorithm builds an ensemble of

iTree over data. In the evaluation stage, for any value x the mean h(x) in ensemble of iTree is computed. In the following sections the average h(x) is used to calculate the anomaly score.

2.3.3.2. Isolation Forest Algorithm [32]

The output of the iForest algorithm is an anomaly score. In short, the anomaly score is average h(x) in iForest normalized by the average path of unsuccessful searches in a Binary Search Tree (BST). In the following part, the individual components of the anomaly score formula are presented.

Average h(x) of the unsuccessful search in BST for the data set of size *i* is:

$$c(i) = \begin{cases} 2H(i-1) - \frac{2(i-1)}{i} & for\ i > 2 \\ 1 & for\ i = 2 \\ 0 & otherwise \end{cases} \qquad 2.5$$

where:

H (*x*) = harmonic number estimated as ln(*x*) + 0.5772156649 (Euler's constant),

As mentioned in the height of iTree is limited so as to manage memory requirements. Formula *c*(*i*) is used to estimate the tree height in cases, where iTree is not able to isolate the point. This is done especially for dense clusters of normal points.

Anomaly formula:

$$s(x, N) = 2^{\frac{-E(h(x))}{c(N)}}. \qquad 2.6$$

with:

x = any row in the data

N = number of rows in the data

E(*h*(*x*)) = mean of *h*(*x*) in ensemble

- The anomaly score is interpreted as follows:
    - if instances return *s* very close to 1, then they are definitely anomalies,
    - if instances have *s* much smaller than 0.5, then they can be quite safely regarded as normal instances,
    - if all the instances return *s* around 0.5, then the entire sample does not have any distinct anomalies.

2.3.4. Support Vector Machine for Binary Classification

Support Vector Machines (SVM) [28] are supervised learning models that find an optimal hyperplane separating two classes in a high-dimensional feature space. By maximizing the margin between classes, SVMs tend to generalize well. Linear SVMs can handle large feature sets, especially when the data are linearly separable after some transformation. Kernel SVMs extends this to nonlinear decision boundaries but at greater computational cost.

The case of a Linear SVM, where the score function is still linear and parametric, will first be introduced, in order to clarify the concept of margin maximization in a simplified context. After that by introducing a kernel the SVM will be made non-Linear and non-parametric [34].

SVM shows its significant advantages on both sparable problems (linear separable problems and non-linear separable problems) and non-separable problems, all will be covered in this section: [33]

- **Case of Linearly separable**

For example, let's say we have been offered some training data with some people's weight, height and their gender, then we want to make use of them to predict the unknown gender data. As shown in (Figure 2.10) these two types of points represent Male and Female, where we can see in (figure 2.11) lines that divide the space into two regions, and we can easily notice that the black solid line would be the optimal line. Which maximizes the margin between itself and the nearest points of each class.

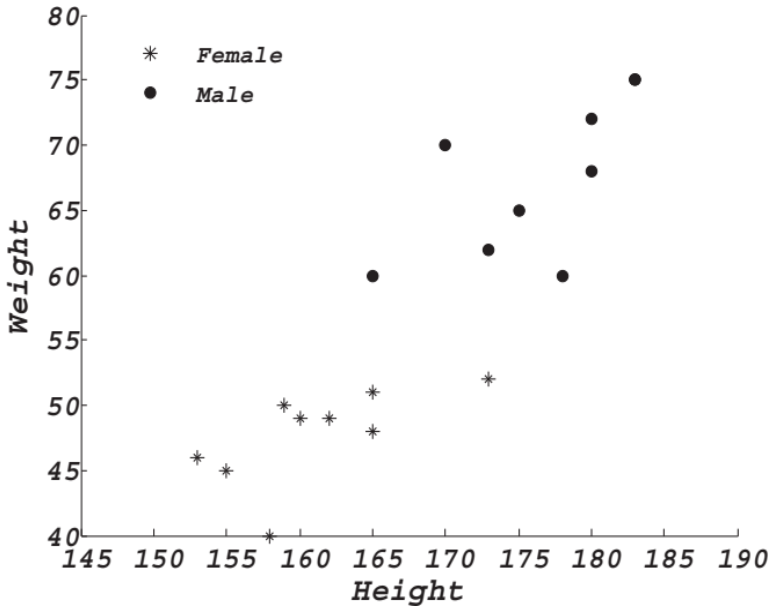


Figure 2.10: Points representing Males Females [33].

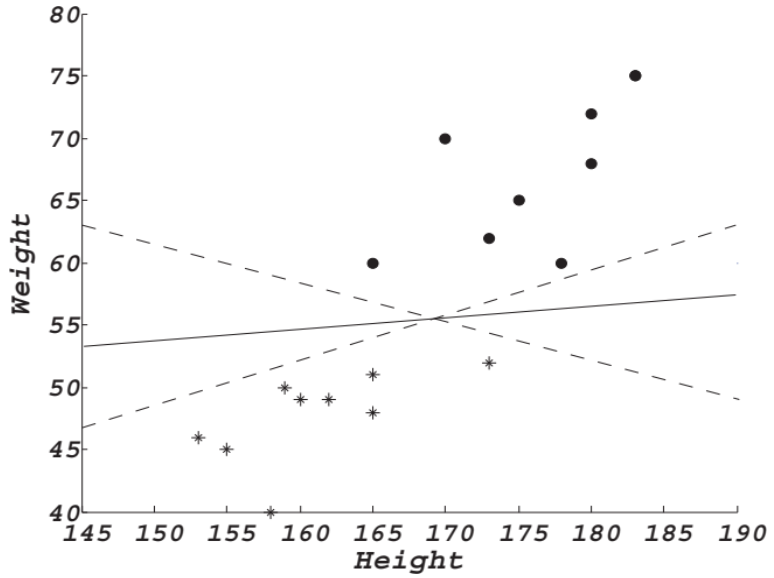


Figure 2.11: Classification Lines [33].

SVM extends the two-dimensional linear separable problem to multidimensional, and aims to seed the optimal classification surface, which we call ***THE OPTIMAL HYPERPLANE*:**

$$w^T x + b = 0 \qquad 2.7$$

W: weight vector.

b: threshold.

The relation between x_i and $f(x_i)$ can be defined as:

$$f(x) = w^T x + b. \qquad 2.8$$

Seeding the optimal hyperplane is equivalent to maximal the distance between the closest vectors to the hyperplane. Define the Euclidean distance between the nearest points and the hyperplane $f(x)$ as:

$$r = \left|\frac{f(x)}{\|w\|}\right| \qquad 2.9$$

$f(x)$: Functional margin

Assume $f(x)$ between the nearest points and the hyperplane is 1 as the (Figure 2.12) shows. The assumption was accompanied with a constraint condition

$$y_i (w^T x_i + b) \geq 1, \; i = 1, 2, \ldots, n, \qquad 2.10$$

which implies that all training data are on the two hyperplane or behind them and the training data on the hyperplane are called support vectors (SVs). Thus, the margin between the parallel bounding planes d can be defined as:

$$d = 2r = \frac{2}{\|w\|} \qquad 2.11$$

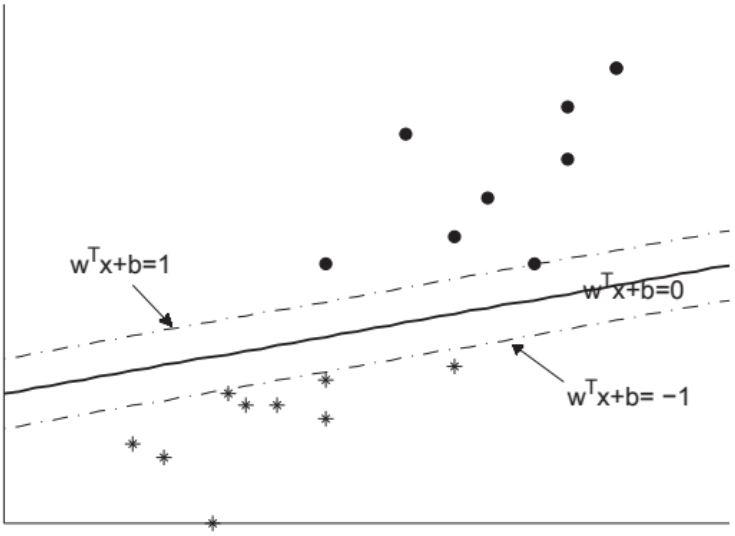


Figure 2.12: Optimal Classification Line [33].

Thus, the objective function can be presented as:

$$min\frac{1}{2}\|w\|^2\ s.t.\ y_i(w^T x_i + b) \geq 1, i = 1,2,\dots,n. \qquad 2.12$$

- **Case of Non-Linearly Separable**

As shown in in (figure 2.13) there is no line that can classify the two classes well, only curves, and this is where SVM shows its superiorities for nonlinearly problems, which takes the way of mapping the input vectors in low dimension feature space to a high dimension feature space to find a suitable line separating hyperplane.

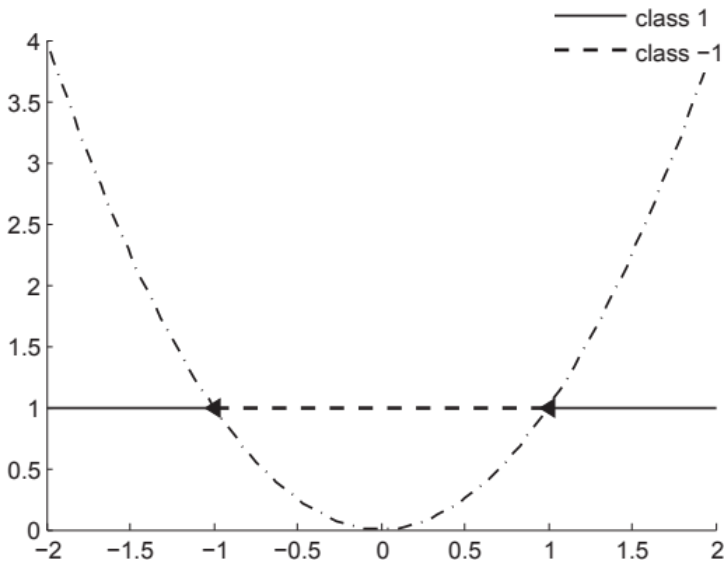


Figure 2.13: Non-linearly Separable [33].

- **Case of non-separable**

Lastly, some points belong to positive class may be counted as negative class or we can say that there is no hyper plane that is able to separate the points of different categories accurately.

These error points are usually regarded as noise which can be ignored by humans but not by machines for machines can't deal with error points like humans do.

In summary, SVM is a powerful classifier, which is suitable for any case of classification with the same decision function, high classification accuracy and small computation [33].

2.3.4.1. Effects of SVM and Kernel Parameters [35]

SVM has a set of parameters different from hyperplane large-margin which we call hyperparameters. The soft margin constant, *C,* and any parameters the kernel function may depend on (width of Gaussian Kernel or degree of a polynomial Kernel), all this will be illustrated with its effect on the decision boundary of an SVN using two dimensional examples.

- Soft-Margin Constant we see in (Figure 2.14) that for a large value of *C*, a large penalty is assigned to errors/margin errors.

Where in the left panel we see the two points closest to the hyperplane affect its orientation making it closer to several other data points, When C is decreased in the right panel those points become margin errors and the hyperplane's orientation changes giving us a much larger margin for the rest of the data.

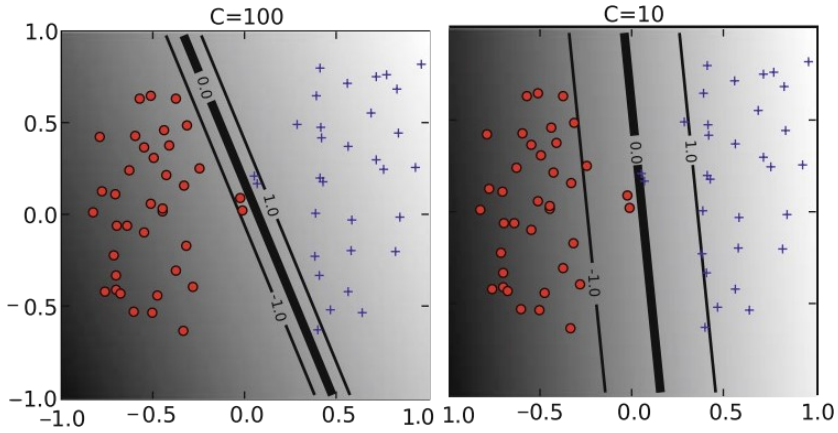


Figure 2.14: The effect of soft-margin constant C [35].

- Decision boundary, the degree of the polynomial Kernel and the width parameter of the Gaussian Kernel control the flexibility of the result in (Figure 2.14) where we find linear kernel that means we have the lowest degree, which is not good for a nonlinear separation.

After that we have a degree 2 polynomial and is flexible enough to discriminate between two classes with a sizable margin.

The degree 5 polynomial has similar decision boundary but with greater curvature.

- Gaussian Kernel, when γ is large, the value of the discriminant function is essentially constant outside the close proximity of the region where the data are concentrated, see bottom right panel in (Figure 2.15). In this regime of the γ parameter, the classifier is clearly overfitting the data.

As seen from the examples in (Figure 2.14 and 2.15), the parameter γ of the Gaussian kernel and the degree of polynomial kernel determine the flexibility of the resulting SVM in fitting the data. If this complexity parameter is too large, overfitting will occur (bottom panels in Figure 2.15).

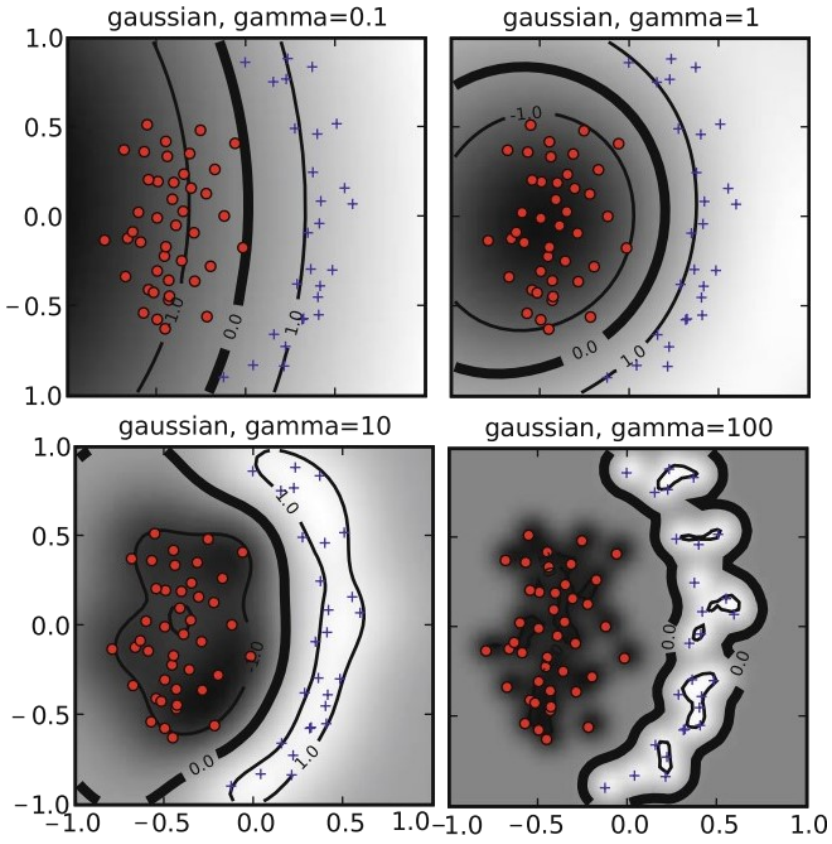


Figure 2.15: The effect of Gaussian Kernel [35].

2.3.4.2. Advantages of SVM [34]

Every classification method has its own pros and cons, and their effectiveness depends largely on the nature of the data being analyzed. Support Vector Machines (SVMs) are particularly beneficial when dealing with data that is irregular or lacks a clear distribution pattern, an issue often encountered in financial distress or insolvency prediction. In these situations, SVMs offer a powerful alternative to traditional methods, especially when working with financial ratios that may not fit neatly into conventional classification models. Here are several reasons why SVMs are advantageous:

- **Flexible Decision Boundaries:** By using kernel functions, SVMs allow for flexible decision boundaries that aren't limited to linear separations. The model doesn't

require the same functional form across all data points, this flexibility reduces the need for manually transforming each problematic variable.

- **No Need for Explicit Feature Transformation:** The kernel trick used in SVMs performs an implicit non-linear transformation of the data, allowing for better separability without requiring assumptions about how to transform the variables beforehand. This makes the process more efficient and less reliant on expert judgment.
- **Strong Generalization Ability:** When SVM parameters like the regularization parameter C and kernel-specific parameters (e.g., r for the Gaussian kernel) are carefully selected, the model tends to generalize well to new data. This robustness is especially important if the training data is biased or not fully representative.
- **Unique, Stable Solution:** SVMs solve a convex optimization problem, which means there's only one optimal solution. Unlike neural networks, which may get stuck in local minima and produce different results depending on the initial weights or sample variations, SVMs are more stable and reliable across different datasets.
- **Similarity-Based Classification:** By using kernels like the Gaussian (RBF) kernel, SVMs place more emphasis on how similar different companies are in terms of their financial ratios. This means that when classifying a new company, the algorithm compares its data to the most relevant support vectors (i.e., examples from the training set that are most similar), leading to more meaningful and accurate classifications based on structural similarity.

2.4 Conclusion

In this chapter we have introduced the basics of machine learning algorithms especially supervised and unsupervised methods, then we have detailed the principal algorithms used in our work, from the PCA, Isolation Forest to SVM.

Chapter 3: Experimental results

3.1 Introduction

In this chapter, we present a structured explanation of a MATLAB-based workflow that combines unsupervised and supervised machine learning techniques. The rationale behind each algorithmic choice is examined, relevant mathematical foundations are derived, and practical considerations for implementation are presented. Finally, evaluation metrics and visualization strategies are described before concluding with a list of key references.

We begin by preprocessing the raw data, estimating noise levels to filter out unreliable spectral bands. We then perform PCA to reduce dimensionality, balancing information retention with computational efficiency.

An Isolation Forest is trained on a 1% random sample of the reduced-dimensional data to compute initial anomaly scores. These scores yield a binary anomaly map (0 = normal,

1 = anomalous), which serves as pseudo-labels for a Support Vector Machine (SVM).

The SVM is trained and then applied across the entire image to generate a refined anomaly prediction.

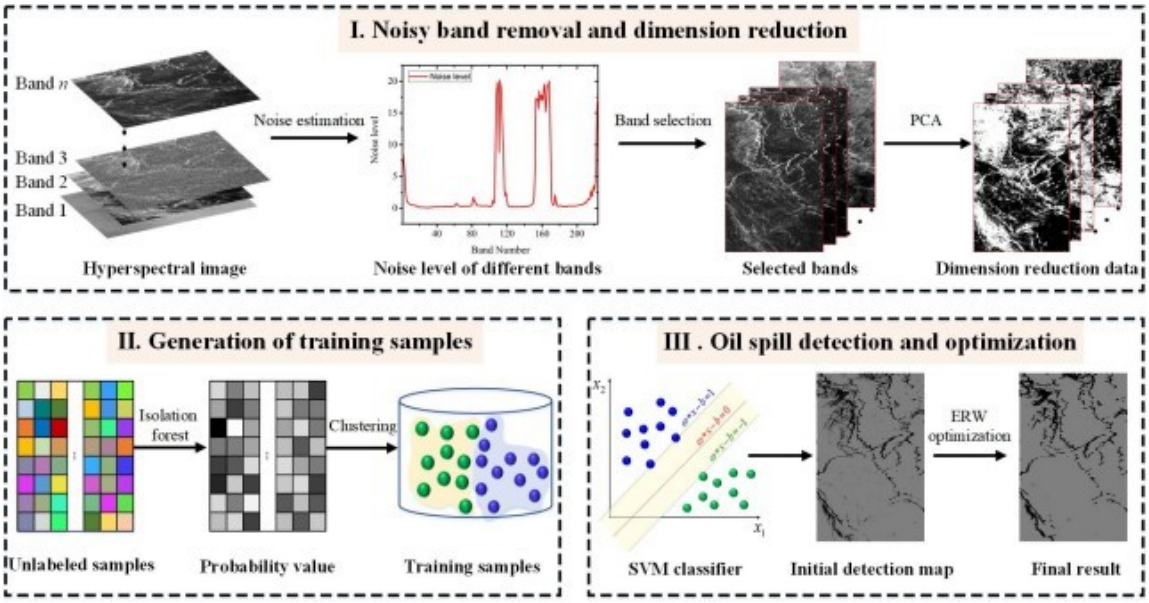


Figure 3.1: The flowchart of the proposed unsupervised oil spill detection method [1].

Finally, we compute performance metrics precision, receiver operating characteristic (ROC) curves, and area under the curve (AUC) and visualize both the reference and predicted anomaly maps.

3.2 Data Sets

3.2.1 Oil Spill Location

The area where we study the Oil spill incident is Located in the Gulf of Mexico, North America continent, around *25° N 90° W*. Known as the biggest environmental incident in US history occurring on April 20, 2010.

More than 757 million liters of raw oil are released at that location.

At this moment is where hyperspectral data is going to be playing a critical role for monitoring and cleaning up the oil spill by detecting the oil spill region providing rich spectral information from the visible to the infrared spectrum.

3.2.2 HOSD Database

A large-scale hyperspectral database was created with AVIRIS sensors from different test sites. This database was called Hyperspectral Oil Spill Database (HOSD), Which was the first public oil spill detection dataset and is freely available for research purposes, this availability helped develop different ground breaking approaches for oil spill detection.

What makes HOSD so special and different from other oil spill datasets used in other publications is the wide distribution, large coverage and large amount of data. Field experts labeled oil spill areas pixel by pixel. The reference maps of all studied sample images are manually annotated by using the ENVI (Environment for Visualizing Images software), as well as the datasets have been processed with atmospheric correction model in ENVI 5.3 software before oil spill detection.

(Table 1) lists some features of the HOSD, spectral coverage is from 365nm to 2500nm.

Table 3.1. Some features of the HOSD [1].

Data	Spatial size	Resolution	Fight time
GM1	1200*633	7.6m	5/17/2010
GM2	1881*693	7.6m	5/17/2010
GM3	1430*691	7.6m	5/17/2010
GM4	1700*691	7.6m	5/17/2010
GM5	2042*673	7.6m	5/17/2010
GM6	2128*689	8.1m	5/18/2010
GM7	2302*479	3.3m	7/09/2010
GM8	1668-*550	3.3m	7/09/2010
GM9	1643*447	3.2m	7/09/2010
GM10	1110*675	7.6m	5/17/2010
GM11	1206*675	7.6m	5/17/2010
GM12	869*649	7.6m	5/06/2010
GM13	1135*527	3.2m	7/09/2010
GM14	1790*527	3.2m	7/09/2010
GM15	1777*510	3.3m	7/09/2010
GM16	1159*388	3.2m	7/09/2010
GM17	1136*660	7.6m	5/17/2010
GM18	1047*550	3.3m	7/09/2010

3.3 Noise Estimation and Band Removal

Our objective is to estimate the noise level σ in each band and remove bands whose noise exceeds a threshold.

Hyperspectral images are often corrupted by noise due to sensor imperfections, calibration errors, and random photon fluctuations during capture. This noise degrades image quality and reduces the accuracy of critical tasks like object detection and land-cover classification. Currently, many researchers manually remove the noisiest spectral bands before analysis.

However, this approach is inefficient, hyperspectral images contain hundreds of bands, making manual inspection tedious.

To solve this problem, we propose an **automatic band selection method** that **quantifies noise** in each band using a Gaussian statistical model then f**iltering out severely noisy bands** using an adaptive threshold.

Real-world noise often follows a Gaussian (normal) distribution, making this model a natural choice.

We estimate noise levels by applying a **Laplacian mask** (below), which amplifies noise-induced pixel variations:

$$M = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix} \quad 3.1$$

The function estimate noise typically computes the standard deviation of local differences along spatial dimensions [15]. Let

$$\sigma n = \sqrt{\frac{\pi}{2}\frac{1}{6(I_W-2)(I_H-2)}}\sum_{i,j}|I_n(i,j) * M|, n = 1,2,..,I_N \quad 3.2$$

Where: I_W and I_H stand for the spatial dimensions, and I_N is the total number of spectral channels.

Bands with high σ contain little useful signal (dominated by noise).

Compute the average noise across all 224 bands:

$$\mathrm{Sn} = \{\mathrm{I_n}, \text{if } \sigma\mathrm{n} < \frac{1}{\mathrm{I_N}}\sum_{\mathrm{n}} \sigma\mathrm{n}\ \emptyset\ , \text{Otherwise} \quad 3.3$$

Any band with $\sigma_n \geq s_n\tau$ is discarded.

This ensures **automatic, data-driven filtering** without manual intervention. Cleaner bands improve downstream tasks like oil spill detection while saving time [1].

3.4 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a popular method for reducing the dimensionality of hyperspectral images. Since these images contain massive amounts of data often with redundant spectral bands PCA helps by compressing the information into fewer key components while keeping the most important details needed for tasks like classification and detection.

As an unsupervised technique, PCA works by analyzing the entire image to find the most meaningful spectral patterns. This not only lowers computational costs but also enhances classification accuracy, which is especially useful when labeled training data is scarce.

However, research shows that PCA may have little effect on detection performance when the target (e.g., an oil spill) has a spectral signature similar to its background. In such cases, the key features needed for detection might not stand out clearly in the reduced PCA components.

So basically, our objective is to reduce the dimensionality of X by projecting onto its principal components, retaining directions of maximal variance.

The first step is to reshape the 3D data cube into a 2D matrix where each row corresponds to a pixel's spectral signature (pixel × band) and ensure numerical validity.

Input: $X \in \mathbb{R}^{(P \times nbands)}$ ($P = nrows \times ncols$).

Output: score the projection of X onto its principal components.

- **Choice of Number of Components**

Although all 224 PCs can be computed, downstream algorithms will use the first Npca =100 PCs. The first 100 PCs typically capture > 99% of the variance in hyperspectral data [19]. Reducing from 224 of bands to 100 components mitigates the "curse of dimensionality" and accelerates anomaly detection algorithms.

After pre-processing, the next step is to identify potential oil spill regions by detecting anomalies in the hyperspectral data.

3.5 Anomaly Detection Using Isolation Forest

The Isolation Forest (iForest) algorithm is an unsupervised machine learning method designed to efficiently detect anomalies by exploiting the fundamental principle that abnormal data points are few in number and inherently different from normal instances. Unlike traditional anomaly detection approaches that rely on computationally expensive distance or density measurements, iForest operates by isolating anomalies through a series of random partitions in the feature space, making it particularly suitable for processing large and complex datasets like hyperspectral remote sensing imagery.

The algorithm consists of two key stages: training and anomaly scoring. In the training phase, the *iForest* model constructs an ensemble of Isolation Trees (*iTrees*), which are binary trees built using a random partitioning process.

Each iTree is generated by recursively splitting the dataset along randomly selected features and threshold values until all instances are isolated. Due to their distinct spectral characteristics, anomalies such as oil spills tend to separate much faster than normal pixels (e.g., water or oil) because they require fewer splits to be distinguished from the majority of the data. This results in significantly shorter path lengths within the trees for anomalous instances compared to normal ones.

Once the forest is built, the algorithm proceeds to the scoring phase, where each pixel in the hyperspectral image is evaluated based on its average path length across all iTrees.

Since anomalies are isolated earlier in the trees, they exhibit shorter average path lengths, which are then converted into anomaly scores.

A higher score indicates a higher likelihood of the pixel being an anomaly, allowing for effective discrimination between oil spills and the background environment.

In summary, the iForest method effectively identifies anomalies that are both rare and spectrally distinct - perfectly matching the needs of hyperspectral anomaly detection.

Since anomaly pixels are typically few in number and differ significantly from background pixels in their spectral characteristics, the isolation concept works well for separating them.

The detection process relies on calculating each pixel's average path length through the isolation trees.

However, the standard iForest approach has limitations, it uses completely random splits at each tree node, choosing arbitrary thresholds that may not optimally separate anomalies from normal pixels. Additionally, during evaluation, the method simply measures path lengths from root to terminal nodes, this becomes problematic when different pixels end up in the same terminal node, they receive identical path lengths despite potentially being different types of anomalies, as shown in (Figure 2.9).
In the implementation of Isolation Forest, we select 1% of the total pixels at random, then train an Isolation Forest to assign anomaly scores.

3.6 Preparing Labels for SVM Training

To bridge the gap between unsupervised anomaly detection and supervised classification, the outputs of the Isolation Forest algorithm specifically, the anomaly scores are transformed into reliable training labels for the subsequent SVM model. This process involves thresholding the anomaly scores to distinguish potential oil spill pixels (anomalies) from the dominant water background, with the threshold selected based on the expected contamination rate (e.g., the top 2% of scores classified as oil). Given that manual labeling of hyperspectral data is labor-intensive and subjective, this automated approach ensures scalability while maintaining detection sensitivity. To minimize label noise, morphological operations such as opening and closing are applied to the binary mask, removing isolated pixels and smoothing irregular spill boundaries.

Additionally, class imbalance is addressed by either weighting the SVM training process or synthetically augmenting oil-labeled samples, ensuring the classifier does not bias toward the more prevalent water class. The final labeled dataset serves as the foundation for training the SVM, enabling precise discrimination between oil and water pixels in the classification stage.

So, to be direct we need to formulate pseudo-labels for a supervised classifier (SVM) based on iForest output, where the Isolation Forest yields an initial unsupervised binary labeling of the random sample. And these labels serve as "weak labels" for training the SVM.

So, the requirements are: At least one normal (0) and one anomalous (1) sample must exist in training data. If not, one must adjust contamination or sample size.

3.7 Final Classification and Oil Spill Mapping

With labeled data generated from Isolation Forest, a Support Vector Machine (SVM) classifier is trained to perform pixel-wise discrimination between oil spills and water, leveraging its ability to handle high-dimensional hyperspectral data through kernel-based separation. The SVM's parameters, including the regularization term (C) and kernel bandwidth (gamma), are optimized via grid search to maximize accuracy on a validation set, with the Radial Basis Function (RBF) kernel selected for its effectiveness in capturing non-linear spectral relationships.

The classifier's output is a binary segmentation mask, which is further refined using spatial post-processing techniques such as median filtering to reduce salt-and-pepper noise and improve boundary coherence.

The resulting classified pixels are then projected onto geographic coordinates, generating an interpretable oil spill map that quantifies spill extent and distribution critical for environmental impact assessments and mitigation efforts. This map can be integrated into GIS platforms for real-time monitoring, providing actionable insights for disaster response teams.

3.8 Postprocessing (Median Filtering)

Reduce spurious isolated misclassifications (salt-and-pepper noise) by applying a 2D median filter to the binary map.

For each pixel (i, j), replace the predicted label with the median of the labels in a local 3×3 neighborhood (default).

Enforce spatial coherence: anomalies often occupy contiguous pixels; an isolated single-pixel anomaly is less likely to be true.
Median filtering smooths small “blips” without blurring edges.

3.9 Performance Metric Computation

In this training process we used an i5-8250U 1.60GHz with 8GB of RAM and 2GB MX150 Graphics Card, Windows 11 64 bits, while using MATLAB 2024a, we also used a training ratio for isolation forest of (1%) random samples for anomaly detection, where the scores yield a binary anomaly map (0= normal, 1= anomaly) and contamination fraction of (2%), which serves as a pseudo-labels for (SVM), the support vector machine is trained using kernel-based learning to classify data by maximizing the margin between classes in high dimensional feature space. Trained linear SVM on the PCA-reduced iForest-labeled data then it is applied to the entire set of pixels identifying it as either oil or water using the trained model.

3.9.1 Confusion Matrix Definitions

- **True Positive (TP):** Pixels correctly identified as anomalies.
- **False Positive (FP):** Pixels incorrectly flagged as anomalies.
- **True Negative (TN):** Pixels correctly identified as normal.
- **False Negative (FN):** Missed anomalies.

3.9.2 Derived Metrics

a) Detection Precision (DP):

$$DP = \frac{TP}{TP+FP}. \qquad 3.4$$

Measures the proportion of predicted anomalies that are correct.

b) True Positive Rate (TPR, a.k.a. Sensitivity or Recall):

$$TPR = \frac{TP}{TP+FN}. \qquad 3.5$$

c) False Positive Rate (FPR):

$$\text{FPR} = \frac{\text{FP}}{\text{FP+TN}} \qquad 3.6$$

3.10. Visualization of Results

The obtained results are summarized below. The calculated detection precision (DP) is shown in (Table 3.2) and (Figure 3.2). As well as the visualization results shown in (Figures 3.3, 3.4 and 3.5) where the result is the prediction map compared to the reference map.

Figure 3.2:

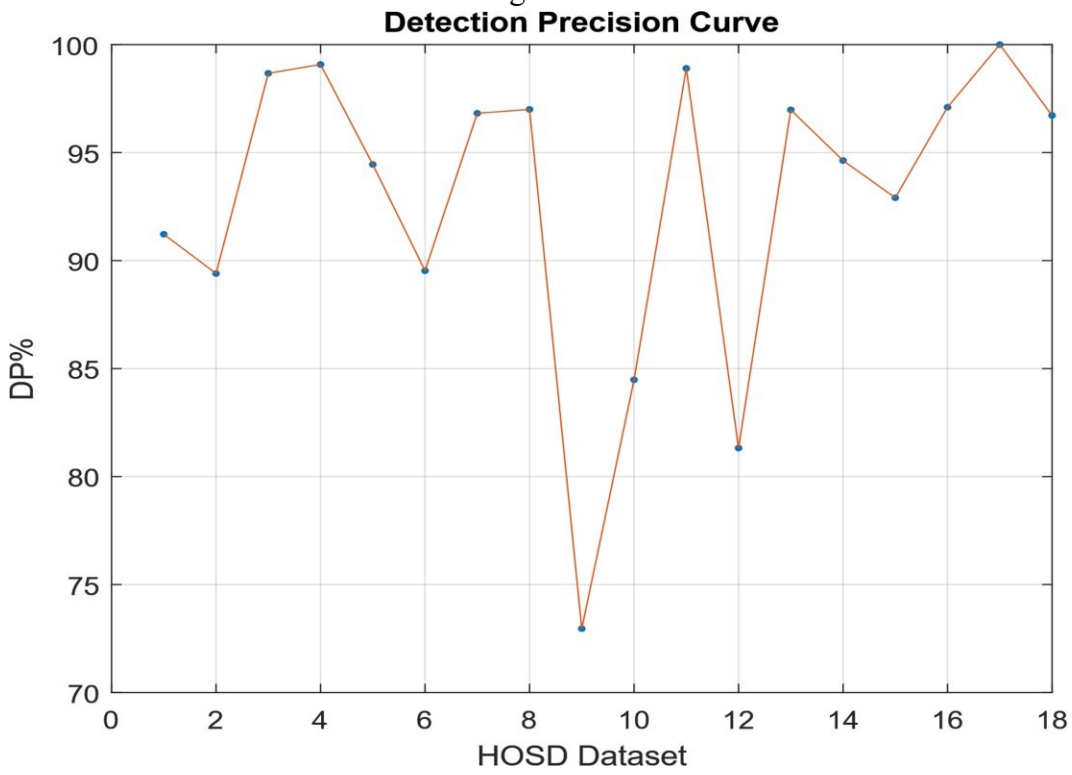


Table 3.2. Numerical results of oil spill detection

	DETECTION PRECISION (DP)	**AREA UNDER CURVE(AUC)**
GM01	91.22%	91.75%
GM02	89.4%	96.53%
GM03	98.67%	90%
GM04	99.08%	92.42%
GM05	94.45%	92.14%
GM06	89.53%	90.86%
GM07	96.82%	77.19%
GM08	97.00%	91.70%
GM09	72.96%	73.11%
GM10	84.48%	90.01%
GM11	98.90%	81.22%
GM12	81.32%	92.63%
GM13	96.98%	68.69%
GM14	94.63%	79.56%
GM15	92.91%	93.93%
GM16	97.10%	92.40%
GM17	100%	65.49%
GM18	96.72%	75.57%

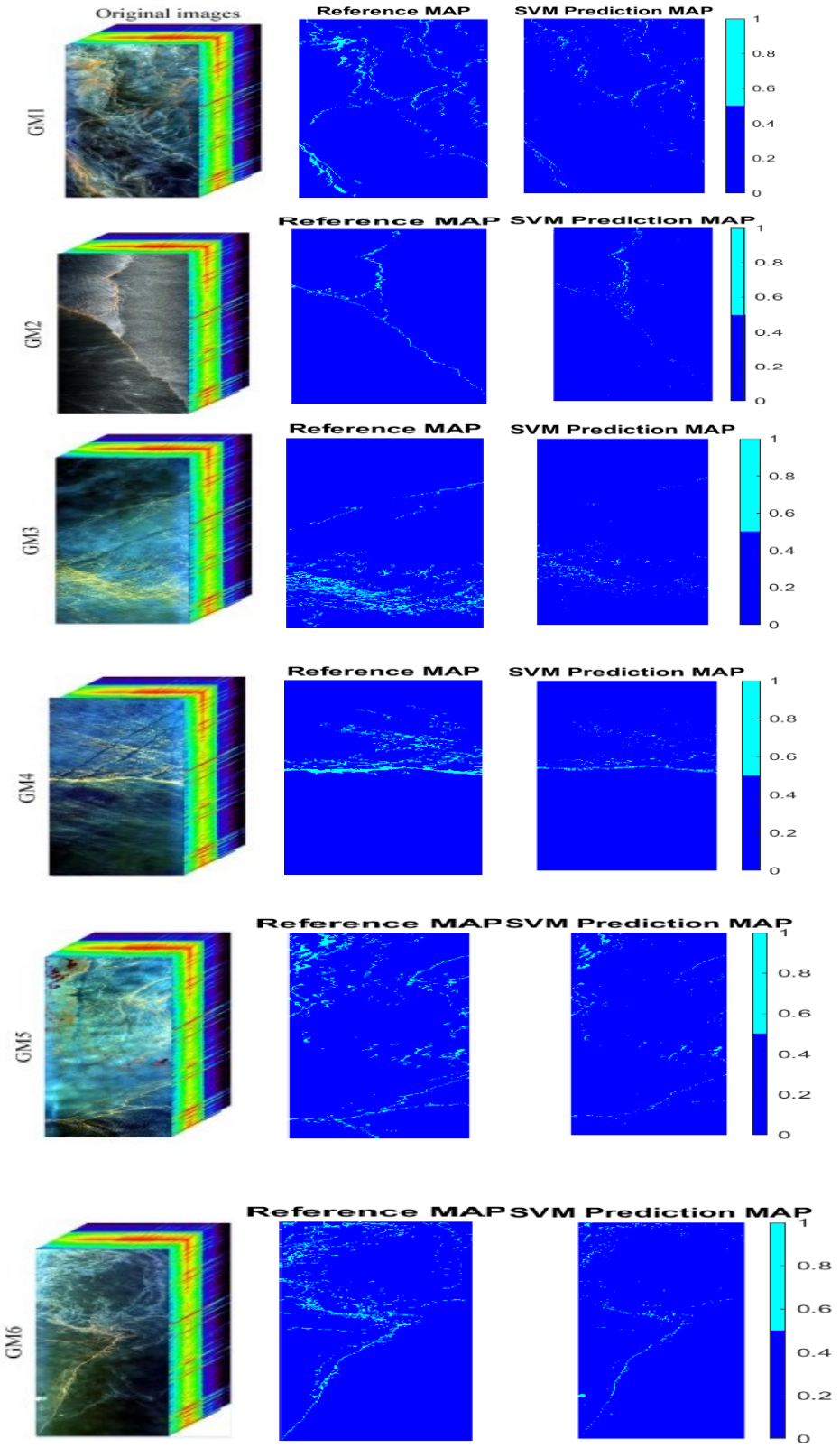


Figure 3.3: The original images, reference map and detection result From GM01 to GM06.

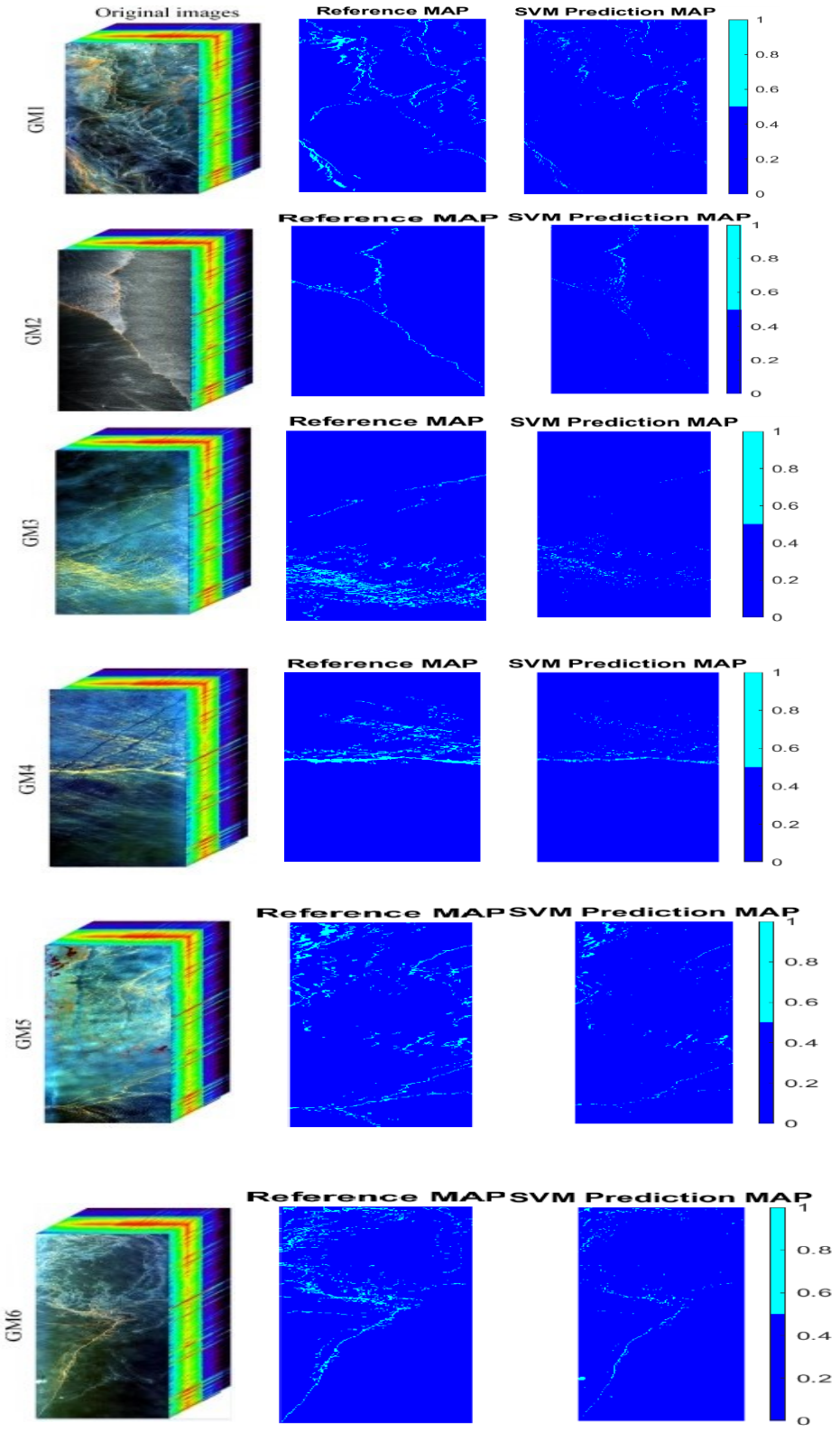


Figure 3.4: The original images, reference map and detection result From GM07 to GM12

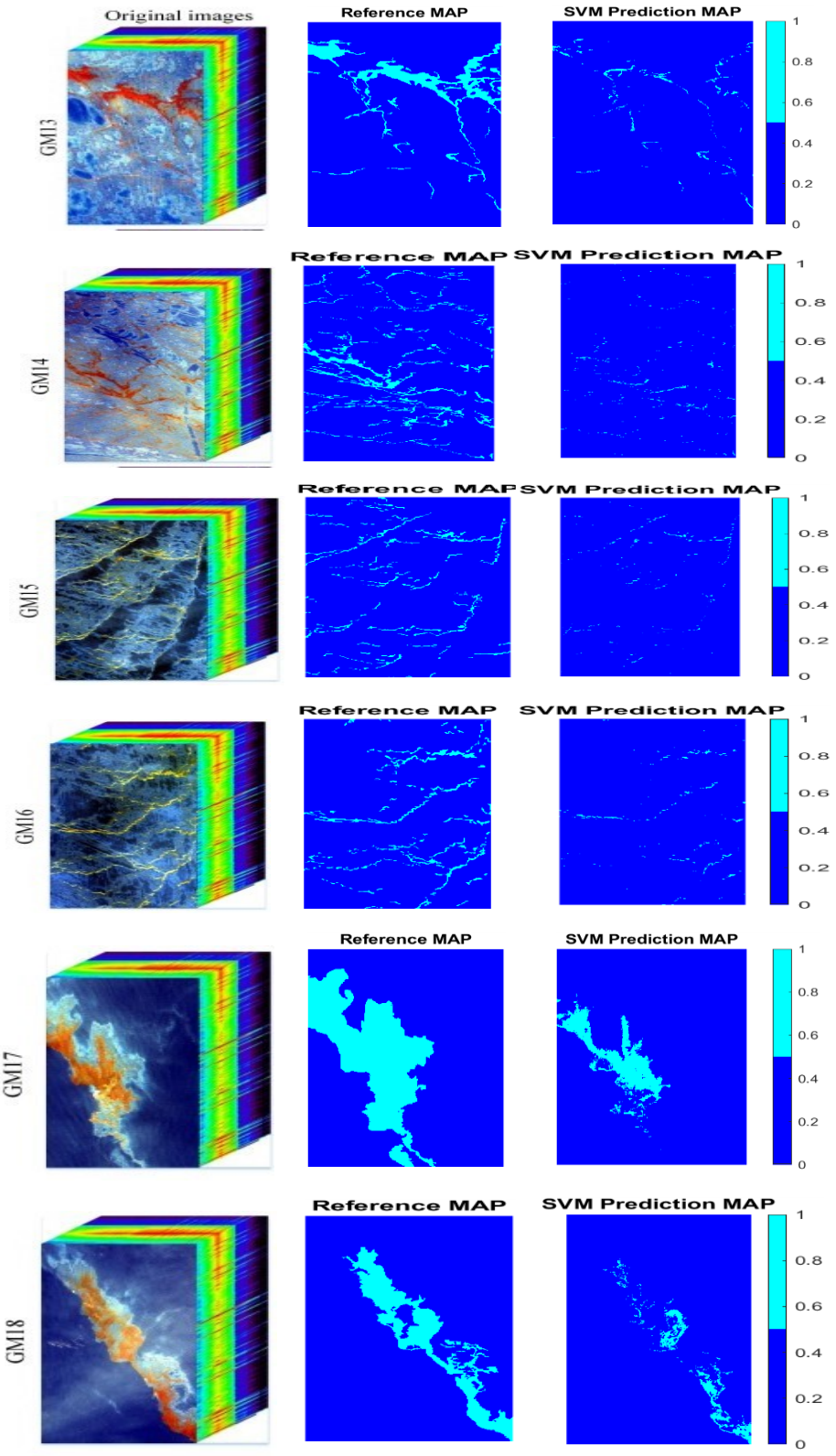


Figure 3.5: The original images, reference map and detection result From GM13 to GM18.

3.11 Discussion

To evaluate the effectiveness of our proposed method for oil spill detection, we grabbed several state-of-the-art methods adopted for comparison such as:

- a scalable exemplar-based subspace clustering (SESC) method [46].
- an unsupervised method derived from rank-two nonnegative matrix factorization (R2NMF) [47].
- a detection approach based on low-rank and sparse matrix decomposition (LRSMD) [48].
- A detection method based on kernel isolation forest (KIF) [49]
- For the supervised oil spill detection method, a PCA- based minimum distance (PCAMD) detection method [50].
- The last approach is an isolation Forest-Guided Unsupervised Detector, using KPCA [1].

All these information and implementations of each approach was found in the publication of journal mentioned in [1].

The results shown in (Table 2) by comparison shows that both LRSMD and R2NMF struggle in detecting oil from water where the precession is very low for LRSMD and below 50% for R2NMF, we see that KIF and PCAMD as well as SESC are a bit better for oil spill detection with acceptable precision through the images but still the Mean is not surpassing 72%, the only method that comes close to ours is the KPCA method with great percentage through the images and even a great Mean of 85.51% but it can be seen that our method produces the highest detection accuracy compared to other approaches on the HOSD database.

Table 3.3. The DP results obtained by all considered approaches on the HOSD [1].

DP	SESC	R2NMF	LRSMD	KIF	PCAMD	KPCA	Our Method
GM01	**96.59%**	11.61%	1.46%	70.80%	57.21%	96.15%	91.22%
GM02	39.41%	14.97%	10.30%	79.53%	74.53%	**96.54%**	89.4%
GM03	98.58%	90.32%	16.24%	74.58%	91.85%	92.90%	**98.67%**
GM04	**99.73%**	50.21%	8.15%	68.54%	75.72%	97.21%	99.08%
GM05	**98.56%**	54.99%	8.26%	73.75%	66.52%	88.15%	94.45%
GM06	46.79%	5.17%	0.29%	80.46%	71.71%	76.26%	**89.53%**
GM07	62.27%	2.62%	0%	51.48%	94.22%	76%	**96.82%**
GM08	96.39%	65.97%	1.69%	70.03%	87.91%	85.35%	**97.00%**
GM09	24.27%	0.14%	0.08%	62.41%	88.07%	60.51%	**72.96%**
GM10	27.58%	45.24%	0%	41.81%	73.62%	**85.26%**	84.48%
GM11	78.13%	54.77%	10.43%	86.06%	66.56%	96%	**98.90%**
GM12	42.44%	27.77%	12.16%	83.08%	69.85%	**92.24%**	81.32%
GM13	87.83%	95.76%	0,81%	38.46%	95.83%	94.04%	**96.98%**
GM14	**99.01%**	89.71%	3.63%	51.66%	95.58%	83.41%	94.63%
GM15	**97.02%**	68.18%	4.55%	39.42%	79.97%	87.12%	92.91%
GM16	70.81%	69.93%	3.74%	32.53%	85.49%	89.44%	**97.10%**
GM17	71.12%	25.26%	0.06%	67.93%	42.89%	51.48%	**100%**
GM18	50.26%	19.04%	0%	79.73%	69.84%	91.15%	**96.72%**
MEAN	71.49%	43.98%	4.55%	64.01%	77.08%	85.51%	**92.89%**

a) Advantages

- **Scalability:** PCA reduces data dimensionality, making downstream algorithms (Isolation Forest, SVM) tractable even on large images.
- **Flexibility:** The combination of unsupervised (Isolation Forest) and supervised (SVM) approaches allows the model to bootstrap from minimal prior knowledge.

- **Localization Accuracy:** SVM refines the anomaly map by learning decision boundaries in the reduced-dimensional space.

- **Parameter Control:** Contamination fraction in Isolation Forest and percentage of pixels sampled are adjustable to tune sensitivity.

b) Limitations

- **Dependence on Contamination Parameter:** If the contamination fraction is set too low or too high, the initial pseudo-labels may be poor, compromising SVM training.
- **Assumption of Linear Separability:** A linear SVM may not perfectly separate anomalies from background if their distributions are highly nonlinear in PCA space. Kernel SVMs could be considered but at higher computational cost.
- **Spatial Context Neglected in Early Stages:** Both PCA and Isolation Forest treat each pixel's spectrum independently. Incorporating spatial features (texture, local neighborhoods) might improve robustness [30].

c) Impact of Parameter Choices

- **Number of PCA Components (Npca)**
  - Too few components may discard discriminative spectral information (increase FN).
  - Too many components slow down subsequent algorithms without proportional gain.
- **Sampling Ratio (1% of pixels)**
  - Larger samples yield more robust iForest boundary but increase runtime.
  - Smaller samples risk missing rare anomalies entirely.
- **Isolation Forest Contamination Fraction (0.02)**
  - Setting contamination = 2% implies expecting 2% anomalies in the sample. If true anomaly proportion is much lower, many false positives labeled in training; if higher, may miss anomalies.

3.12 Conclusion

This chapter details each component of the algorithm used for hyperspectral anomaly detection that combines PCA, Isolation Forests, and SVMs. We begin by estimating per-band noise and discarding highly noisy bands to enhance signal-to-noise ratio. We then apply PCA to reduce the dimensionality of the hyperspectral cube, capturing the lion's share of spectral variance in a reduced basis. A small random sample (1% of total pixels) is used to train an Isolation Forest, which assigns initial anomaly scores.

Our results demonstrate the superior performance of this integrated approach, consistently achieving higher detection precision compared to several state-of-the-art methods, including SESC, R2NMF, LRSMD, KIF, PCAMD, and even a similar KPCA-based method. The mean detection precision of 92.89% achieved by our method on the HOSD database stands as a testament to its effectiveness in accurately identifying oil spill regions.

We also discussed its limitations, such as the dependence on the contamination parameter and the assumption of linear separability in the SVM. Future work could explore the integration of spatial features and the optimization of parameter choices (e.g., number of PCA components, sampling ratio, and SVM kernel tuning) to further enhance robustness and accuracy. This research significantly contributes to the advancement of remote sensing techniques for environmental monitoring, offering a powerful tool for rapid and precise oil spill detection.

Conclusion

Conclusion

This thesis presents a complete framework for unsupervised oil spill detection in hyperspectral images using a combination of Isolation Forest and Support Vector Machines (SVM). Beginning with an overview of hyperspectral imaging and its significance in remote sensing applications, we explored the core concepts behind spectral data acquisition, sensor technologies, and image processing techniques.

Our methodology introduces a noise estimation and band elimination step to improve spectral quality, followed by Principal Component Analysis (PCA) to reduce dimensionality while preserving essential variance. Using the Isolation Forest algorithm, we conducted unsupervised anomaly detection to identify oil spill regions based on their spectral uniqueness. The outputs of the Isolation Forest served as pseudo-labels to train an SVM classifier, resulting in a robust oil spill classification map.

Experimental evaluation using the Hyperspectral Oil Spill Dataset (HOSD) demonstrated the effectiveness and scalability of the proposed approach. Performance metrics such as precision and AUC indicated high detection accuracy across multiple test cases. Overall, this research offers a promising contribution to the field of environmental monitoring by providing an efficient and automated oil spill detection system. Future work can explore the integration of spatial features, advanced neural networks, and real-time processing for enhanced performance and generalization

References

References

[1] Puhong Duan, Xudong Kang, Pedram Ghamisi, ‘’ Hyperspectral Remote Sensing Benchmark Database for Oil Spill Detection with an Isolation Forest-Guided Unsupervised Detector’’, journal of LATEX Templates, September 30, 2022.

[2] https://www.alaska.edu/epscor/

[3] Adolfo Martínez-Usó, José Martínez Sotoca., ‘’ From Narrow to Broad Band Design and Selection in Hyperspectral Images’’, Conference Paper in Lecture Notes in Computer Science · June 2008.

[4] Anuja Bhargava, Ashish Sachdeva, Kulbhushan Sharma, Mohammed H. Alsharif, Peerapong Uthansakul, Monthippa Uthansakul, ‘’Hyperspectral imaging and its applications’’, Review article in Helyion, June 2024.

[5] MUHAMMAD JALEED KHAN, HAMID SAEED KHAN, ADEEL YOUSAF, KHURRAM KHURSHID, ASAD ABBAS, ‘’ Modern Trends in Hyperspectral Image Analysis’’, Review article in IEEEAccess, March 12, 2018.

[6] Landsat Science, “Landsat 1 (ERTS-1),” [Online]. Available: https://landsat.gsfc.nasa.gov

[7] A. F. H. Goetz et al., “Imaging spectrometry for Earth remote sensing,” *Science*, vol. 228, no. 4704, pp. 1147–1153, 1985.

[8] J. B. Solomon and A. F. H. Goetz, “Imaging spectrometry and hyperspectral remote sensing: Evolution and application,” *Remote Sensing of Environment*, vol. 113, S110–S122, 2009.

[9] R. O. Green et al., “Imaging spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS),” *Remote Sensing of Environment*, vol. 65, no. 3, pp. 227–248, 1998.

[10] R. Lu and Y. Chen, “Hyperspectral imaging for detection of internal defects in apples,” *Postharvest Biology and Technology*, vol. 18, no. 2, pp. 215–226, 2000.

[11] P. Gowen, C. O’Donnell, P. Cullen, G. Downey, and J. Frias, “Hyperspectral imaging – an emerging process analytical tool for food quality and safety control,” *Trends in Food Science & Technology*, vol. 18, no. 12, pp. 590–598, 2007.

[12] B. M. Nicolaï et al., “Nondestructive measurement of fruit and vegetable quality by means of NIR spectroscopy: A review,” *Postharvest Biology and Technology*, vol. 46, no. 2, pp. 99–118, 2007.

[13] https://aviris.jpl.nasa.gov/

[14] Mohammed Almuqati, Fatimah Sidi, Siti Nurulain Mohd Rum, Maslina Zolkepli, ‘’Challenges in Supervised and Unsupervised Learning: A comprehensive Overview’’, ResearchGate, Vol. 14, no. 4, August 2024.

[15] Reed, I. S., & Yu, X. (1990). Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(10), 1760–1770.

[16] Chen, Y., & Jæger, O. (2001). Robust estimation of covariance matrices for anomaly detection in hyperspectral imagery. *Proceedings of the 2001 International Geoscience and Remote Sensing Symposium*, 1, 224–226.

[17] Chang, C.-I. (2003). Hyperspectral Data Exploitation: Theory and Applications. *Wiley*.

[18] Chen, Y., Li, J., & Su, W. (2009). A kernel target detection algorithm for hyperspectral imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2(1), 63–69.

[19] Jolliffe, I. T. (2002). Principal Component Analysis (2nd ed.). *Springer*.

[20] Hyvärinen, A., & Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, 13(4–5), 411–430.

[21] Green, A. A., Berman, M., Switzer, P., & Craig, M. D. (1988). A transformation for ordering multispectral data in terms of image quality with implications for noise removal. *IEEE Transactions on Geoscience and Remote Sensing*, 26(1), 65–74.

[22] Fukunaga, K. (1990). Introduction to Statistical Pattern Recognition (2nd ed.). *Academic Press*.

[23] Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. *Proceedings of the 2008 IEEE International Conference on Data Mining*, 413–422.

[24] Yousra Chabchoub, Maurras Ulbricht Togbe, Aliou Boly, Raja Chiky, ‘’ An In-Depth Study and Improvement of Isolation Forest’’, Publication in IEEEAccess, January 18, 2022.

[25] Knorr, E. M., & Ng, R. T. (1998). Algorithms for mining distances-based outliers in large datasets. *Proceedings of the 24th International Conference on Very Large Data Bases*, 392–403.

[26] Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF: identifying density-based local outliers. *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, 93–104.

[27] Fei Tony Liu, Zhi-Hua Zhou. ‘’ Isolation Forest’’, Confrence Paper in ResearchGate, January 2009.

[28] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.

[29] Bioucas-Dias, J. M., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P., & Chanussot, J. (2012). Hyperspectral remote sensing data analysis and future challenges. *IEEE Geoscience and Remote Sensing Magazine*, 1(2), 6–36.

[30] Chen, Y., Nasrabadi, N. M., Tran, T. D., & European, G. (2016). Hyperspectral image subspace representation and its application in large-scale processing. *IEEE Transactions on Geoscience and Remote Sensing*, 54(10), 6079–6092.

[31] Bruno Pelletier, '' On the statistical properties of the isolation forest anomaly detection method'', July 15, 2024.

[32] Bc. Adam Valenta, ''Anomaly detection using Extended Isolation Forest'', Thesis, 2021.

[33] Huibing Wang, Jinbo Xiong, Zhiqiang Yao, Mingwei Lin, Jun Ren, '' Support Vector Machin'', Research Survey, 10th, July 2017.

[34] Auria, Laura; Moro, Rouslan A, '' Support Vector Machines (SVM) as a technique for solvency analysis'', Working Paper in ECONSTOR, 2008.

[35] Asa Ben-Hur, '' Support Vector Machines'', A User's Guide in ResearchGate, January 2010.

[36] J. Richards and X. Jia, *Remote Sensing Digital Image Analysis*, 5th ed. Berlin, Germany: Springer, 2013.

[37] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[38] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2001.

[39] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York, NY, USA: Springer, 2002.

[40] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York, NY, USA: Wiley, 2001.

[41] https://www.machinelearningplus.com/machine-learning/principal-components-analysis-pca-better-explained/.

[42] D. Manolakis and G. Shaw, "Detection algorithms for hyperspectral imaging applications," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 29–43, Jan. 2002.

[43] J. M. Bioucas-Dias et al., "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote*

Sens., vol. 5, no. 2, pp. 354–379, Apr. 2012.

[44] B. W. Hapke, *Theory of Reflectance and Emittance Spectroscopy*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[45] https://www.machinelearningplus.com/machine-learning/principal-components-analysis-pca-better-explained/