Håvard Hjelmeseth
Sofie Olsvik Johansen

Unsupervised Band Selection with SPPF: A Spectral-Spatial Fuzzy Clustering Approach

Master's thesis

NTNU

Norwegian University of
Science and Technology

Håvard Hjelmeseth
Sofie Olsvik Johansen

Unsupervised Band Selection with SPPF: A Spectral-Spatial Fuzzy Clustering Approach

NTNU

Norwegian University of
Science and Technology

Abstract

Acquiring a hyperspectral image (HSI) involves capturing multiple spectral bands within a specific wavelength range. However, processing HSIs is computationally demanding due to the immense amount of data. Band selection (BS) methods are crucial in mitigating the challenge of high dimensionality and redundancy of HSI data. While several unsupervised BS techniques are available, such as those based on clustering, there is a need for further exploration into leveraging spatial information. To address this gap, an improved hybrid of the particle swarm optimization (PSO) algorithm with fuzzy clustering (FCM) is proposed, incorporating spatial information. This unsupervised BS technique aims to enhance HSI classification by addressing the challenges of high dimensionality and redundancy in hyperspectral data.

The novelty of this approach, referred to as Superpixel PSO-FCM (SPPF), lies in the extension of PSO-FCM by leveraging both spectral and spatial information. Different distance measures incorporating spatial information are presented, and their effect on classification accuracy and efficiency is explored through experiments. To better select a subset of bands that effectively represents the entire HSI, while minimizing redundancy, the thesis proposes combining the distance measures with different cluster-representative selection techniques. Performance evaluation on three HSIs showcases the promising results of the proposed method.

ii

Sammendrag

Anskaffelsen av et hyperspektralt bilde (HSI) innebærer å fange flere spektrale bånd innenfor et bestemt bølgelengdeområde. Behandling av HSIer er imidlertid krevende på grunn av den enorme mengden data. Båndseleksjons (BS) metoder er avgjørende for å håndtere utfordringen med høy dimensjonalitet og redundans av HSI data. Selv om det finnes flere unsupervised BS metoder, er det behov for ytterlige forskning knyttet til utnyttelse av romlig informasjon. For å adressere dette foreslås en forbedret hybrid av partikkelsvermoptimalisering (PSO) med fuzzy clustering (FCM), som inkorporerer romlig informasjon. Denne unsupervised BS-teknikken tar sikte på å forbedre HSI-klassifisering ved å håndtere utfordringene med høy dimensjonalitet og redundans i hyperspektral data.

Nyvinningen med denne tilnærmingen, referert til som Superpixel PSO-FCM (SPPF), ligger i utvidelsen av PSO-FCM ved å utnytte både spektral og romlig informasjon. Forskjellige avstandsmålinger som inkorporerer romlig informasjon blir presentert, og deres effekt på klassifiseringsytelse og effektivitet utforskes gjennom eksperimenter. For bedre å velge et delsett av bånd som effektivt representerer hele HSIer, samtidig som redundans minimeres, foreslår denne oppgaven å kombinere avstandsmålingene med forskjellige metoder å velge bånd fra en gruppering. Evaluering av ytelsen på tre forskjellige HSIer viser de lovende resultatene av den foreslåtte metoden.

Preface

This thesis emerged as the outcome of a master's project in artificial intelligence at the Norwegian University of Science and Technology. We would like to thank our supervisor, Professor Pauline Haddow, for providing exceptional guidance and detailed feedback throughout this project.

Håvard Hjelmeseth, Sofie Olsvik Johansen
Trondheim, June 9, 2023

Contents

List of Figures

x

List of Tables

Chapter 1

Introduction

This chapter introduces the motivation for the thesis topic and research goals (Section 1.1 and 1.2) for a novel bio-inspired algorithm to perform unsupervised band selection for hyperspectral images. It also presents the research method (Section 1.3), a structured literature review protocol (Section 1.4), and the thesis structure (Section 1.5).

1.1 Motivation

Remote sensing using hyperspectral imaging is a powerful tool for collecting and analyzing data about the Earth's surface. By capturing and measuring the reflectance at different wavelengths, hyperspectral imaging facilitates the identification and mapping of various surface features, including vegetation, minerals, and water bodies. It is also an effective technique for monitoring environmental changes [Jia et al., 2021] and conducting land cover classification [Patro et al., 2021]. This makes it a valuable tool for scientists, engineers, and other professionals who need to monitor and analyze the environment.

However, due to the high dimensionality of hyperspectral images (HSI), extracting something meaningful from the data can be costly. To address this challenge, band selection (BS) methods have emerged as an effective approach to reduce the number of spectral bands in a HSI while retaining essential information. These methods identify and select the most relevant bands from the original image, reducing the computational burden for subsequent image-processing tasks. Determining the relevance of bands can be accomplished by considering either spectral or spatial information in the HSI, relying on measures of information content and redundancy between bands. Despite the numerous hyperspectral BS methods that have been proposed, the investigation into utilizing spatial infor-

mation still is limited.

The PSO-FCM BS algorithm, proposed in Zhang et al. [2017], uses a hybrid of particle swarm optimization and Fuzzy C-means clustering to select bands to be used for classification. This thesis builds on the central ideas of PSO-FCM and proposes the integration of both spatial and spectral information to improve performance when applied to HSI classification.

1.2   Goals and Research Questions

The goal and research questions of this work:

Goal *To improve classification accuracy and efficiency on HSIs through unsupervised band selection by applying a superpixel-enhanced PSO-FCM hybrid algorithm.*

This thesis aims to investigate ways of potentially improving classification accuracy and runtime through an improved hybrid of Particle Swarm Optimization and Fuzzy C-means clustering. The suggested improvements involve the incorporation of superpixel data to allow the hybrid algorithm to utilize spatial information.

Research Question 1 *How does pre-removal of noisy bands influence the achieved classification accuracy of unsupervised BS with the superpixel-enhanced PSO-FCM algorithm?*

Research Question 2 *How can PSO-FCM utilize superpixel information and different divergence measures in order to achieve the best classification accuracy?*

Research Question 3 *What is the influence of different distance measures incorporating superpixel information on computational efficiency?*

Research Question 4 *How does the performance and efficiency of the superpixel-enhanced PSO-FCM algorithm compare to the baseline and other state-of-the-art BS methods?*

1.3   Research Method

The research method of this thesis has been a systematic review of related work (described in Section 1.4), the design of our hybrid algorithm based on the knowledge gained and aspects of existing work, and an analytical evaluation of how our proposed model has fulfilled the research goal.

1.4 Structured Literature Review Protocol

Figure 1.1 presents a comprehensive exploration of potential research topics that ultimately guided the focus of this thesis. Selections that were pursued are indicated in green, while other explored but not further pursued fields of inquiry are denoted in white. Evidently, as seen in Figure 1.1, initial investigations centered around the broad field of evolutionary computation and its diverse applications. Concepts such as leveraging Artificial Immune Systems for fake news discrimination and employing Ant Colony Optimization for network control and design were among the initial considerations.

PSO emerged as an area of collective interest, highlighted in green in Figure 1.1. This interest consequently narrowed the investigative scope to two prospective thesis directions.

One proposition entailed investigating the application of Social PSO for modeling climate change. This approach aimed to understand potential predispositions in individuals to enhance their contributions and sacrifices for future generations' benefit. The second proposition involved exploring how PSO could enhance classification accuracy in HSIs from the field of remote sensing through unsupervised BS. Given the compelling nature of the application domain of PSO within remote sensing and the potential of PSO in BS methods, the latter was chosen for further exploration, as shown in Figure 1.1.

After choosing the initial topics, a literature review was conducted on particle swarm optimization (PSO) for unsupervised BS in HSIs. The process began with the identification of the subject of interest. Subsequently, a comprehensive search for relevant studies and articles was conducted using various databases and resources, including:

- Google Scholar

- IEEE Xplore

- Research Gate

- Web of Science

- SpringerLink

In order to ensure the relevance and quality of the studies included in the review, inclusion and quality criteria were established. The Inclusion criteria required that studies satisfy at least one of the following:

- The study or article is relevant to band selection.

- The study or article is relevant to particle swarm optimization.

- The study or article concerns evolutionary computation.

The studies were then evaluated on the following Quality Criteria:

- The study or article must be well-written and clearly present the research question, methods, results, discussions, and conclusions.

- The study or article must be based on large and representative sample data.

- The study or article must report results that are reliable and statistically significant.

The first phase of the literature review identified the use of superpixels and fuzzy c-mean clustering as potentially interesting directions for further research. The second phase of the literature review focused specifically on BS techniques that use these methods and applied the inclusion and quality criteria to the studies identified. Each study underwent reading and evaluation, with key findings and limitations being noted. The results and limitations of the individual studies were then compared, highlighting common themes and gaps in the existing research. This synthesis contributed to a more comprehensive understanding of the current state of knowledge on the topic and highlighted potential directions for future research.

The primary keywords utilized in the literature search are listed in Table 1.1, differentiating between those employed in the first phase and those employed in the second phase:

Keywords (First Phase)	Keywords (Second Phase)
Particle swarm optimization (PSO)	Superpixel
Unsupervised	Fuzzy c-mean clustering
Band selection	PSO-FCM
Hyperspectral Image	

Table 1.1: Keywords used for literature search.

1.5   Thesis Structure

The thesis is structured as follows. Chapter 1 gives an introduction to the topic, providing an overview of the research methodology and its aims. Chapter 2 provides the background theory necessary to understand the concepts behind this thesis. Chapter 3 presents the state of the art, discussing existing solutions and

more recent approaches within the related fields. The chapter is divided into four sections: *Hyperspectral Band Selection*, *Band Selection Techniques*, *Particle Swarm Optimization*, and *PSO-FCM*, with the final section providing design details necessary for the subsequently proposed BS model. Chapter 4 describes the proposed model applied for BS, including the novelty of this work. Chapter 5 presents the experiments and results of the research, discussing the methods used and the findings. Finally, Chapter 6 evaluates the research and draws conclusions, discussing the implications of the results and any future work.

It should be noted that some of the content from Chapters 1 and 3 is adapted from older versions delivered as part of the pre-project conducted during the fall semester of 2022. This content has been redeveloped based on feedback received from our supervisor.
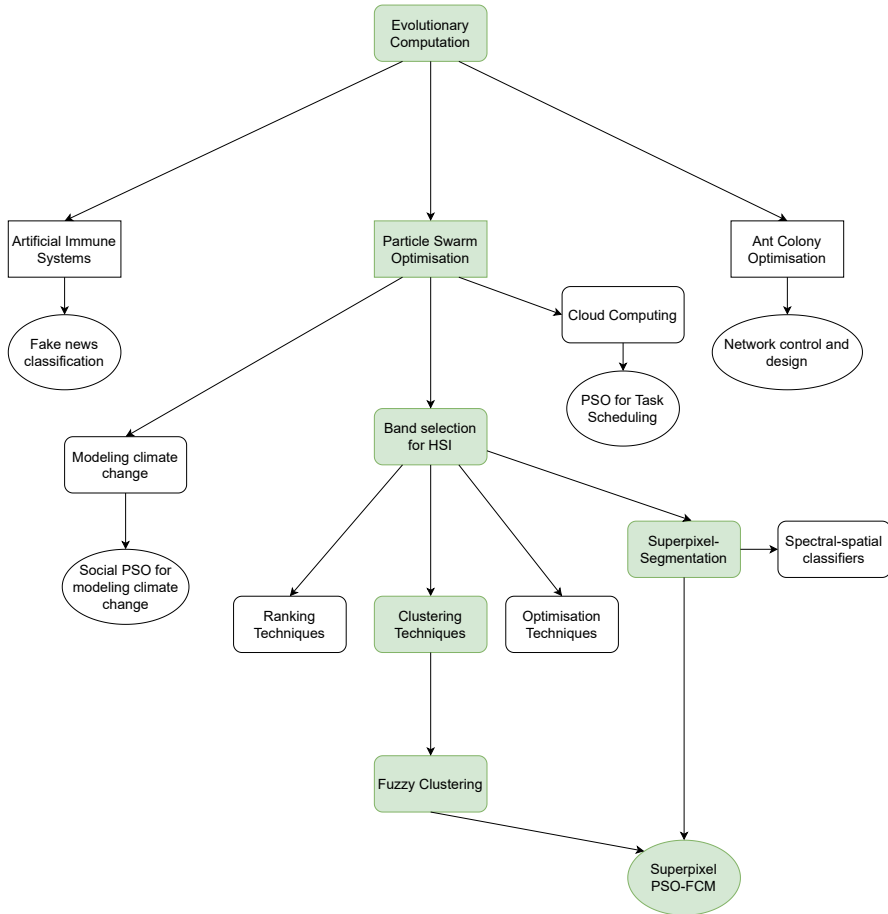
Figure 1.1: Initial topic investigation.

Chapter 2

Background Theory

This chapter provides an overview of the background theory required to understand the concepts presented in later chapters. The first section provides the basics of remote sensing (Section 2.1). The following sections present the relevant concepts of information theory (Section 2.2) and image segmentation (Section 2.3). The next section briefly describes evolutionary computation before presenting the genetic algorithm and particle swarm optimization (Section 2.4). The final section describes clustering and provides examples of relevant algorithms to methods presented in later chapters. (Section 2.5)

2.1   Remote Sensing

Remote sensing is the process of obtaining information on objects or phenomena from a distance. It includes domains such as data collection from planets using space probes and the use of sonar technology to collect data from the ocean floor. Most commonly, it refers to the observation and detection of objects on Earth. By utilizing sensor arrays, usually carried by aircraft or satellites, remote sensing technology enables the collection and analysis of information from large land areas.

The information collected through remote sensing can be used to identify and monitor features of the Earth's surface, such as changes in land use, vegetation, and water bodies. As a result, remote sensing can be applied to investigate and monitor issues such as climate change and natural disasters. Therefore, advancements in remote sensing technology and research can impact many other fields of study and address various challenges facing our planet, ranging from environmental issues to urban planning and resource management.

As mentioned, the data is collected from arrays of sensors. These arrays often

consist of multiple electromagnetic sensors that collect radiation reflected from the Earth's surface at different wavelengths. Such remote sensing technology can either be passive or active. As shown in Figure 2.1, passive sensors rely on reflected sunlight as the primary source of radiation. Whereas in active remote sensing, the collector emits energy directed at the target and gathers information on the reflected radiation. RADAR (Radio Detection and Ranging) technology is an example of active remote sensing as it emits radio waves to detect and locate objects.

The light passing through the lenses of both passive and active sensors is dispersed into distinct bands, covering different wavelengths of light. However, the number and width of these bands can significantly differ from one sensor to another.

(a) Passive Remote Sensing   (b) Active Remote Sensing
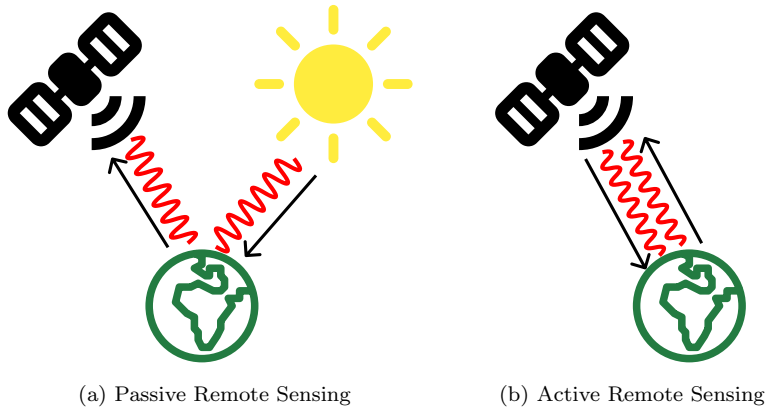
Figure 2.1: Difference between passive and active remote sensing.

2.1.1   Hyperspectral Images

Hyperspectral imaging is a remote sensing technique that captures the spectral signature of an object or scene. It measures the reflection or emission of light across a wide range of contiguous, narrow wavelength bands, allowing researchers to obtain detailed information about the composition and physical properties of the target. The hyperspectral sensors typically collect data from wavelengths between ultraviolet and mid-range infrared.

A distinction is made to differentiate between hyperspectral and multispectral sensors. While they both measure radiation across multiple spectral bands, the number and size of these bands vary. The differences between the two are

illustrated in Figure 2.2, where the resulting spectral bands are highlighted. Multispectral sensors typically collect data in fewer but broader bands, which are often not directly adjacent in terms of wavelengths, as seen in (a). In contrast, hyperspectral sensors typically use many more and narrower bands, as seen in (b). The higher number of bands is equivalent to a continuous spectrum, giving hyperspectral sensors a higher level of detail than multispectral sensors.
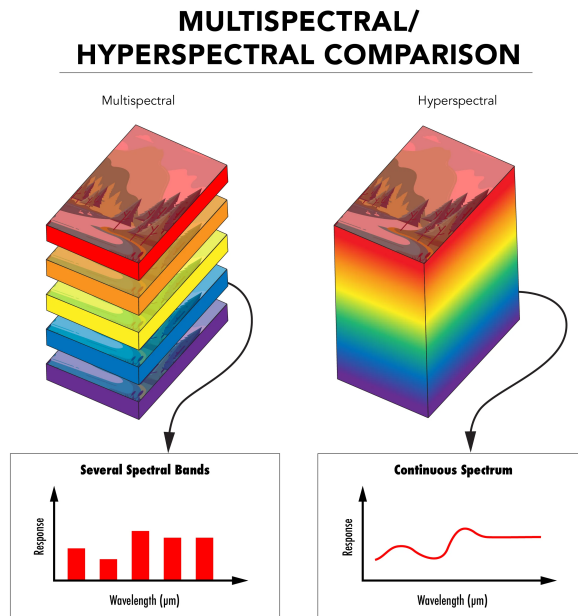
Figure 2.2: Comparison of Multispectral and Hyperspectral Imaging [Optics, 2022].

2.1.2 Classification

The process of assigning pixels in a HSI to a particular class or category based on its spectral properties is referred to as classification. The outcome of this process is known as the classification map. Classification can be divided into two types based on the availability of training samples: *supervised* and *unsupervised*. In supervised classification, representative samples, known as training samples, are used to classify input data for each class. By contrast, unsupervised classification does not require labeled data and identifies patterns and relationships among data points.

HSI classification is an essential step in transforming hyperspectral data into feature information. The features in HSIs are highly dimensional, which can lead to a problem known as the Hughes phenomenon [Hughes, 1968]. Since the classification accuracy decreases as the number of features increases beyond a certain threshold, it requires more labeled data to estimate the function accurately. To tackle this problem, reducing the dimensionality of the feature vector is necessary. Feature vector in this context refers to a multidimensional representation of the spectral information of a pixel or a region of interest in an image, where each dimension corresponds to a specific spectral band. There are two methods to achieve this: feature selection and feature extraction. Feature selection aims to identify a subset of features that retain as much information as possible from the original features. In contrast, feature extraction maps the feature space to a new, lower-dimensional space. In this thesis, classification is employed as an application to verify the performance of a BS model operating on hyperspectral data. Furthermore, this section offers a concise introduction to the selected classifier utilized in the experiments and presents a brief overview of incorporating spatial information within the classifier.

Support Vector Machine (SVM)

Support Vector Machines (SVM) is a widely used supervised method for solving classification problems in machine learning. In the context of SVM classification, the goal is to find the hyperplane that separates the data into different classes with the maximum margin. Figure 2.3 shows the maximum margin hyperplane (illustrated by the red line) after training an SVM classifier with two classes. In essence, the SVM algorithm identifies the optimal line or decision boundary, known as a hyperplane, by locating the closest points between the two classes, referred to as support vectors. In Figure 2.3, the support vectors are illustrated by black dotted lines. The SVM aims to maximize the margin, which is the distance between the support vectors and the hyperplane. The hyperplane with the largest margin is considered the optimal one.

While the SVM classifier is inherently a linear classifier, adaptions have been made to support classification problems with data that is not linearly separable. The *kernel trick* is a technique to transform the data into a higher-dimensional space in which it is linearly separable. Different kernel functions, such as polynomial, radial basis function (RBF), and sigmoid, can be used to achieve this transformation.

The problem of binary classification with SVM involves a set of training vectors, x, which belong to a d-dimensional space. Each training vector x is associated with a target value, either -1 or $+1$. The goal is to find a hyperplane that separates the two classes linearly and without errors. The hyperplane can
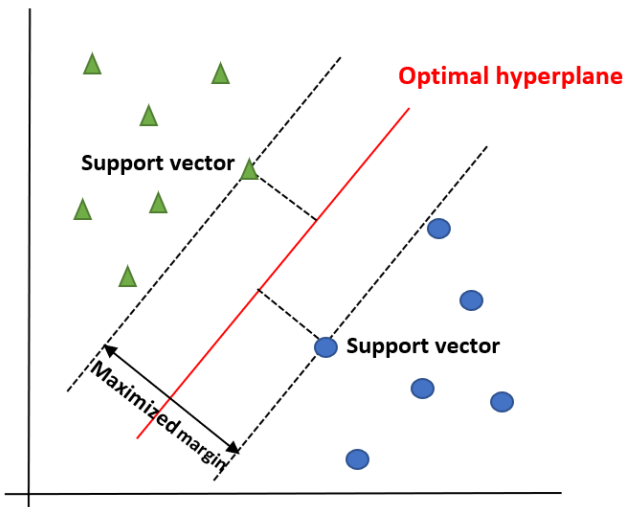
Figure 2.3: Output of SVM Classifier.

be defined as:

$$w^\mathsf{T}x - b = 0, \qquad (2.1)$$

where w is a vector normal to the hyperplane, and b is a bias term. To find the maximum margin hyperplane, SVM utilizes two other hyperplanes:

$$w^\mathsf{T}x - b = 1, \qquad (2.2)$$

and

$$w^\mathsf{T}x - b = -1. \qquad (2.3)$$

The goal of SVM is to find the values of w and b that minimize the classification error and maximize the margin between the two classes.

In addition to the basic binary SVM classifier, other algorithm variations are available that enable it to deal with problems such as multiple categories. SVM can be utilized for multi-class classification by training multiple classifiers, each specialized in different pairs of classes. An example of a common implementation applied for multi-class classification is developed by [Chang and Lin, 2011]. LIBSVM, a library for SVM, implements the "one-against-one" method, which involves training a classifier for every possible pair of classes. When a new data point needs to be classified, each classifier votes on the class they think it belongs to, and the class with the most votes is chosen as the final classification.

Spatial-Spectral Classification

Spatial-spectral classification combines spatial and spectral information to classify objects or features on the Earth's surface. This classification method uses the spatial and spectral characteristics of pixels in HSIs to classify them into different classes or categories. Spatial information refers to the location and arrangement of pixels in the image, while spectral information refers to the intensity values of the different wavelengths of electromagnetic radiation that are captured.

2.1.3   Band Selection

A challenge in hyperspectral imaging involves the typically large number of spectral bands measured. A common strategy to reduce data size and enhance data analysis performance involves selecting a subset of relevant bands from the total hyperspectral information. This process is termed band selection (BS). When data size reduction occurs, maintaining as much significant information from the original image becomes essential. Therefore, selecting bands carrying information relevant to the desired image analysis is necessary.

In certain situations, *a priori* information about the image exists, allowing for the use of this data to select bands. Employing such *a priori* information receives the designation of supervised BS, frequently yielding superior outcomes compared to situations without its use. At times, *a priori* data remains unavailable, necessitating the deployment of alternative measures for choosing suitable bands. This scenario is classified as unsupervised BS.

2.2   Information Theory

Information theory, first introduced by Claude Shannon in 1948 [Shannon, 1948], is the scientific study of the transmission and processing of information. The central idea is quantifying information in a given message or information source. In other words, it provides a measure of information called entropy, which quantifies the amount of randomness in a given information source. Information-theoretic measures can be applied to any message if the message's probability mass function P can be determined. Therefore a method to assess the probability mass function of the image needs to be established to apply information theoretic measures to a HSI.

Regarding BS, information theory is applied to measure the quantity of information in HSIs and the relative information overlap between bands. The normalized grayscale histogram of a band is assumed to be a reliable estimator of the respective band's probability distribution.

2.2.1 Entropy

Entropy is an essential concept in Information Theory. It measures the amount of information in data by quantifying how dispersed the data is over its possible values. A high entropy indicates a maximum data spread, implying high uncertainty since each value is equally likely to occur. In contrast, a low entropy suggests the concentration of the data on a single value, implying less uncertainty since the next value can be predicted with higher confidence.

Entropy represents a random data source's average rate of information production and can therefore work as a measure of randomness present in a spectral band. The greater the entropy, the more information the band conveys. Utilizing the normalized grayscale histogram as an estimator for the probability distribution, the entropy for a spectral band is defined as follows:

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i), \tag{2.4}$$

where X is the random variable representing the spectral band, $p(x_i)$ is the probability (grayscale histogram value) of the i-th bin, and n is the number of bins. The entropy provides a lower bound on the average number of bits required to represent the information in the source, which in this case, is the spectral band.

2.2.2 Kullback–Leibler Divergence

The Kullback–Leibler (KL) divergence, also known as relative entropy, was originally introduced by Kullback and Leibler [1951]. It quantifies the difference between two probability distributions. Given two probability distributions P and Q, the KL divergence between P and Q is defined as:

$$D(P,Q) = \sum_{i} P(i) \log \frac{P(i)}{Q(i)}, \tag{2.5}$$

where $P(i)$ and $Q(i)$ are the probabilities of event i under distributions P and Q. The KL divergence is not symmetric, meaning $D(P,Q) \neq D(Q,P)$. The KL divergence is not a preferred distance metric due to this property. To resolve this problem, the symmetrization of the KL divergence is a common approach. The symmetrized KL divergence, denoted as $D_{sym}(P,Q)$, is defined as:

$$D_{sym}(P,Q) = D(P,Q) + D(Q,P). \tag{2.6}$$

It is important to note that the symmetrized KL divergence will be zero if and only if P and Q represent the same distribution, and it is always non-negative.

For hyperspectral BS, KL divergence is measured in pairs of bands. Therefore given two bands i and j in a HSI, the KL-divergence between them can be defined as:

$$D_{sym}(i,j) = \sum_{n=1}^{N} p_{n,i} \log \frac{p_{n,i}}{p_{n,j}} + p_{n,j} \log \frac{p_{n,j}}{p_{n,i}} \tag{2.7}$$

where N is the number of bins and $p_{n,i}$ and $p_{n,j}$ are the values in the n-th bin of the grayscale normalized histograms of the i-th and j-th band.

2.2.3   Mutual Information

Mutual information measures the amount of information shared between two random variables. It provides a quantitative measure of the dependence between the variables. In the case of hyperspectral BS, the variables refer to the normalized grayscale histograms of two spectral bands. Given two random variables X and Y, the mutual information $MI(X,Y)$ between them is defined as:

$$MI(X,Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{2.8}$$

where $p(x,y)$ is the joint probability distribution of X and Y, and $p(x)$ and $p(y)$ are their marginal probability distributions. The mutual information quantifies the dependence between the two variables, with higher values of mutual information indicating a stronger dependence between the spectral bands.

2.2.4   Disjoint Information

Disjoint information refers to independent and non-overlapping information between two random variables and is calculated by subtracting the mutual information from the joint entropy. Similar to mutual information, in the case of band selection the variables refer to the grayscale histograms of two spectral bands. The disjoint information between two variables, X and Y, is defined as;

$$DI(X,Y) = H(X,Y) - MI(X,Y), \tag{2.9}$$

where $H(X,Y)$ is joint entropy and $MI(X,Y)$ is mutual information. Disjoint information is related to the KL divergence, which quantifies the difference between two distributions instead of their similarity.

2.3 Image Segmentation

Image segmentation lies in the field of image processing and computer vision, and has proved to be a valuable technique in a broad area of research. Some of its applications are medical imaging, object detection, recognition tasks, and content-based image retrieval. Often, image segmentation is used as a pre-processing step for further image analysis.

Image Segmentation describes the separation of objects within a digital image. The image is separated into multiple segments, typically regions containing adjacent pixels representing objects or borders in the image. Each segment is homogenous, containing pixels with similarities concerning characteristics such as color, texture, or light intensity. Image segmentation separates the image into disjunct regions covering the complete image. The number of segments can either be pre-defined or found dynamically, and the segmentation can either be supervised or unsupervised.

To ensure an understanding of image segmentation for this thesis, the subsequent subsections will delve into specific methods that are important for grasping the concepts and techniques related to superpixel segmentation, which will be discussed in detail later in this section.

2.3.1 Thresholding

Thresholding is often considered a simple but effective method of image segmentation. The method segments pixels in a grayscale image into one of two segments, depending on whether the pixel intensity is over a certain threshold T. This threshold can be set manually or found automatically. The segment a pixel at position (i, j) belongs to can be found using the following equation:

$$S_{i,j} = \begin{cases} 1 & \text{if } I_{i,j} > T \\ 0 & \text{if } I_{i,j} \leq T, \end{cases} \quad (2.10)$$

where $I_{i,j}$ represents the intensity of that pixel, and $S_{i,j}$ the segment it belongs to.

Such partitioning typically transforms the image into a binary grayscale image, where every pixel is either 1 or 0. However, it can also be performed with multiple thresholds to segment the image into more than just two classes, as shown in Figure 2.4.

2.3.2 Graph Partitioning Methods

Graph partitioning techniques represent another, more complex, approach to performing image segmentation. In such methods, the image is represented as

| (a) MRI Scan Image | (b) Thresholding using a single threshold | (c) Thresholding using 3 thresholds |
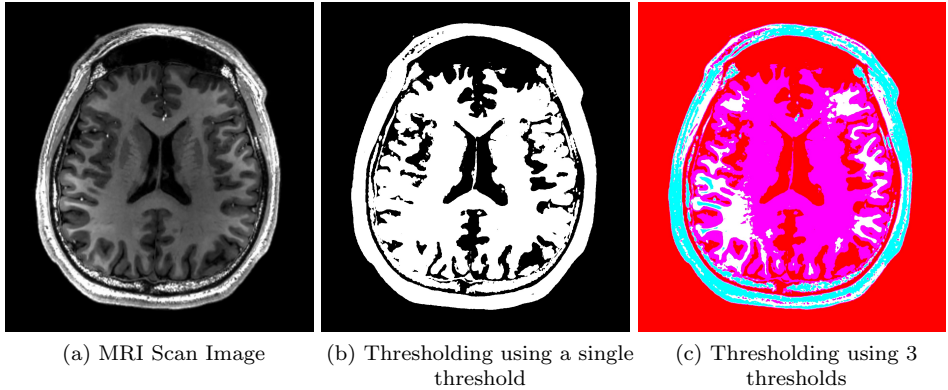
Figure 2.4: Image segmentation using thresholding.

a weighted, undirected graph. Each node represents a pixel in the image, and the edges connecting each node represent how similar two neighboring pixels are based on different possible measures. Graph partitioning methods then partition the graph into multiple smaller graphs, each representing a segment in the final segmentation. There are numerous ways to perform this partitioning, and the most significant difference between such algorithms lies in the segmentation criteria.

Normalized Cuts

Normalized cuts (NCut), first introduced by Shi and Malik [2000], is a commonly used graph partitioning method based on minimizing the normalized cut criterion. The algorithm partitions the graph in such a way that the cut between the segments is minimized while maximizing intra-segment similarity.

The method assumes a graph $G = (V, E)$ that can be partitioned into two disjoint sets A and B, so that $A \cup B = V$. The optimization criterion in NCut is called the *normalized cut* and is defined as:

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)}, \quad (2.11)$$

where $assoc(A, V)$ and $assoc(B, V)$ denote the total connection from nodes in A and B to all nodes in the graph. Furthermore, $cut(A, B)$ represents the degree of dissimilarity between two subgraphs A and B, and is defined as:

$$cut(A, B) = \sum_{u \epsilon A, v \epsilon B} w(u, v), \tag{2.12}$$

where $w(u, v)$ represents the weight of the edge between nodes u and v.

The total connection from nodes in a subgraph A to all nodes in the graph is defined as:

$$assoc(A, V) = \sum_{u \epsilon A, t \epsilon V} w(u, t), \tag{2.13}$$

where $w(u, t)$ represents the weight of the edge between nodes u and t. $assoc(B, V)$ is similarly defined.

2.3.3 Superpixel Segmentation

Superpixel segmentation is a type of image segmentation that uses an over-segmentation to separate the image into regions (termed superpixels) smaller than traditional image segmentation regions. Unlike traditional image segmentation, as seen in Figure 2.5 (b), superpixels are not intended to cover entire objects within the original image in (a). Instead, superpixel segmentation aims to group pixels exhibiting similar characteristics within an object, allowing for variations in superpixel sizes, as shown in Figure 2.5 (c).

(a) Original Image   (b) Image segmentation   (c) Superpixel segmentation
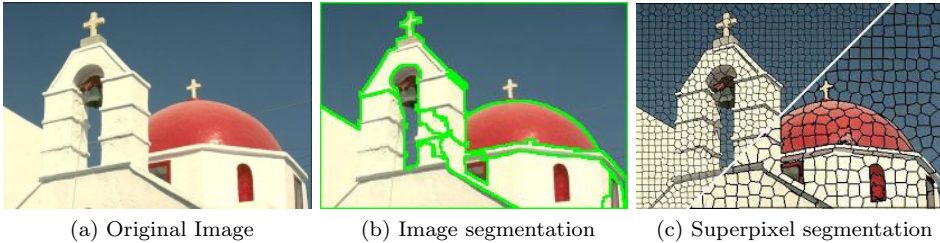
Figure 2.5: Difference between image segmentation and superpixel image segmentation.

Entropy Rate Superpixel Segmentation

Entropy Rate Superpixel Segmentation (ERS), first introduced in Liu et al. [2011], is a graph-based superpixel segmentation algorithm. Similarly to NCut, the algorithm assumes a weighted undirected graph $G = (V, E)$, that can be partitioned into disjoint subsets $\mathcal{S} = \{S_1, S_2, ..., S_K\}$ such that $S_i \cap S_j = \varnothing$ if $i \neq j$ and

$\bigcup_i S_i = V$. Vertices and edges are denoted by v_i and $e_{i,j}$ respectively, with the similarity between them given by the non-negative weight $w_{i,j}$. The method also assumes that every vertex in the graph has a self-loop, with weight defined as $w_{i,i}$.

ERS works by selecting a subset of edges $A \subseteq E$ such that the resulting graph, $G = (V, A)$, contains exactly K connected subgraphs. The algorithm then uses the entropy rate of a random walk on the constructed graph as the criterion to obtain compact and homogeneous clusters and uses a balancing function to encourage clusters of similar size.

The entropy rate of a random walk on the graph G = (V, A) is defined as:

$$\mathcal{H}(A) = - \sum_i \mu_i \sum_j p_{i,j}(A) \log(p_{i,j}(A)), \quad (2.14)$$

where the transition probability $p_{i,j}(A)$ is given by:

$$p_{i,j}(A) = \begin{cases} \frac{w_{i,j}}{w_i} & \text{if } i \neq j \text{ and } e_{i,j} \in A \\ 0 & \text{if } i \neq j \text{ and } e_{i,j} \notin A \\ 1 - \frac{\sum_{j:e_{i,j} \in A} w_{i,j}}{w_i} & \text{if } i = j. \end{cases} \quad (2.15)$$

The sum of incident weights of the vertex w_i is given by:

$$w_i = \sum_{k:e_{i,k} \in E} w_{i,k} \quad (2.16)$$

and the asymptotic distribution μ_i defined as:

$$\mu_i = \frac{w_i}{\sum_{j=1}^{|V|} w_j}. \quad (2.17)$$

The balancing function $B(A)$, which helps maintain similar cluster sizes, is defined as:

$$\mathcal{B}(A) = - \sum_i \frac{|S_i|}{|V|} \log \left(\frac{|S_i|}{|V|} \right) - N_A, i = \{1, \cdots N_A\}, \quad (2.18)$$

where N_A is the number of connected components in the graph, and $|S_i|, |V|$ is the cardinality of S_i and V, respectively.

Combining the entropy rate with the balancing function leads to the objective function that needs to be maximized:

$$\max_A \mathcal{H}(A) + \lambda \mathcal{B}(A), \quad (2.19)$$

where $\lambda \geq 0$ is the weight of the balancing term. The objective function is then greedily optimized to construct the final segmentation.

Simple Linear Iterative Clustering

Simple Linear Iterative Clustering (SLIC), first introduced in Achanta et al. [2012], is a clustering-based superpixel segmentation algorithm that generates superpixels by applying the k-means clustering algorithm based on the relationship between color similarity and spatial distance. The initialization involves converting an image into a feature vector in CIELAB color space (a three-dimensional entity containing the entire range of human color perception) and spatial coordinates. Then pixels are clustered to obtain regular superpixels by constructing a distance measure for the feature vector. The initialization of N superpixel clustering centers of the same size is followed by the computation of the distance D within a $2Q \times 2Q$ block region around the superpixel centers, where Q is computed as the square root of the number of pixels in the image divided by N. The distance measure between the clustering center i and the pixel j within the block is given by:

$$D_{i,j} = \sqrt{\left(\frac{D_{i,j}^{(\text{S})}}{Q}\right)^2 + \left(\frac{D_{i,j}^{(\text{C})}}{\omega}\right)^2}, \tag{2.20}$$

where $D^{(\text{S})}$ represents the spatial distance, $D^{(\text{C})}$ represents the color distance, and ω is the maximum color distance within a given cluster. The spatial position and CIELAB color values of two pixels, denoted by x, y, a, b, and c, can be used to calculate $D^{(\text{S})}$ and $D^{(\text{C})}$ as follows:

$$D_{i,j}^{(\text{S})} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{2.21}$$

$$D_{i,j}^{(\text{C})} = \sqrt{(a_i - a_j)^2 + (b_i - b_j)^2 + (c_i - c_j)^2}. \tag{2.22}$$

Following the initial clustering, the clustering centers undergo iterative updates. These updates are based on the mean values of the distance measures within the corresponding clustering blocks. This process continues until the clustering centers for each pixel reach a stable state and stop changing.

2.4 Evolutionary Computation

Evolutionary computation denotes a group of algorithms for global optimization that is inspired by biological processes such as genetic evolution or swarm behavior. The algorithms typically rely on a population of candidate solutions and iteratively improve through multiple generations. The algorithms are highly adaptable and can produce good solutions for a wide range of optimization problems.

2.4.1   Genetic Algorithm

The genetic algorithm (GA), introduced by Holland [1975], is an optimization
algorithm that iteratively optimizes a population of candidate solutions. As the
name implies, the algorithm is inspired by the biological processes of evolution,
combining operators such as parent selection, gene crossover, and mutation to
converge to good solutions. The genetic algorithm is classified as an Evolution-
ary Algorithm, which usually performs well at approximating solutions to vari-
ous problems. One of the main advantages of using a GA is that it can handle
problems with complex fitness-landscape. Fitness-landscape describes multidi-
mensional spaces where each point represents a possible solution and its corre-
sponding fitness value to a problem.

The genetic algorithm works on a *population* of candidate solutions and itera-
tively improves its solutions throughout multiple *generations*. In each generation,
the entire population is evaluated using a fitness function, and a subset of the
population is then selected for reproduction. Reproduction typically happens by
mixing the chromosomes of two parent individuals to create two new individu-
als (offsprings), which are then subjected to a mutation operator. This process
is repeated for multiple generations until a stopping criterion is met. Stopping
criteria often include: reaching a maximum number of generations, having no
improvement in solutions for a set number of generations, or having found a
sufficiently good solution. The entire process is illustrated in Figure 2.6.

Representation

The genetic representation describes how individuals in the population should
be represented in the form of a *chromosome* or *genotype*. Such a representation
must have a predefined mapping to the *phenotype*/solution. Selecting which
genetic representation to use, largely depends on the problem at hand, but the
optimal solution must be included in the set of possible genomes. One of the most
straightforward genetic representations is a single sequence of binary numbers,
such as $G = 01110101$. Such a representation is often used when solving problems
related to whether to include or not include certain features. For permutation-
or sequence-related problems such as TSP or VRP, an alternative representation
could be a string of characters representing the chosen sequence, such as $G = ADBCE$.

Evaluation of Individuals

For a genetic algorithm to perform optimization, it needs to know what it should
optimize. This is typically referred to as the *fitness function*. The fitness function
evaluates every individual in the population for every generation. It can return
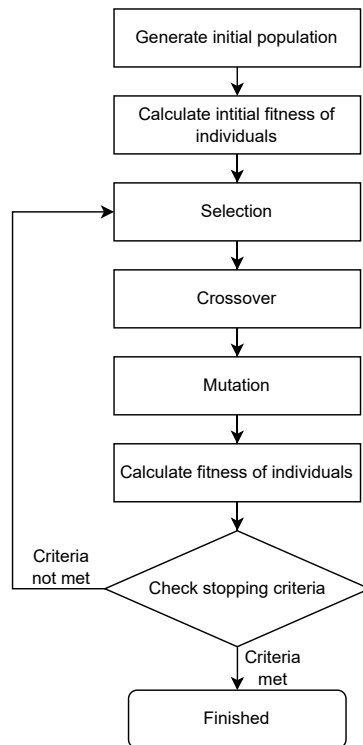
Figure 2.6: Flowchart of Genetic Algorithm.

either positive or negative fitnesses and can be chosen to either be maximized or minimized throughout the generations. If the optimization problem has multiple objectives to be optimized, these can be combined into a single fitness function, in the form of $f = aX + bY$, where X and Y are separate objectives, with balancing coefficients a, b. Alternatively, multi-objective optimization can be performed by considering Pareto dominance in the selection process.

Selection

The selection mechanism in the Genetic algorithm is an integral part of guiding the population toward the optimal solutions. After the evaluation of every individual has been performed, the selection mechanism can utilize the fitness scores to select individuals that are well suited as "parent" individuals to create the next generation. Many selection methods exist, typically designed to favor the selection of individuals with good fitness while simultaneously maintaining

diversity in the population, often through some form of weighted randomness. This procedure is then repeated until the desired number of individuals has been selected for reproduction.

Crossover

The crossover operator decides how the offspring of two parent individuals should be created. Choosing the best crossover operator often depends on the chosen genetic representation and the problem that is being solved.

For binary sequence representations, one-point crossover and k-point crossover are common operators. These select one or k crossover points respectively, and fill each subsequence between those points with the same subsequence from one of their parents randomly. In some cases, like the mentioned TSP or VRP, slightly more complex crossover operations may be needed to retain the good parts of the parent chromosomes and generate feasible solutions. *Partially mapped crossover* (PMX) and *Order crossover* (OX1) are examples of typical crossover operations for permutation representations.

Mutation

The mutation operator is typically designed to make minor changes to a solution, often done to encourage exploration and preserve diversity in the population. The mutation usually occurs after crossover, with a low probability p_m of it happening to each gene/character in the chromosome. For binary representations, this typically means flipping a bit from 0 to 1, or the other way around. In sequence representation, like in the case of TSP, a mutation typically involves swapping the positions of two characters in the sequence. When choosing mutation operators, it is essential to consider how the mutation affects the phenotype, as flipping a single bit can in some cases lead to very significant changes in the resulting solution.

2.4.2   Particle Swarm Optimization

Particle Swarm Optimization (PSO), first introduced by Kennedy and Eberhart [1995], is a computational method for finding the optimal solution to a given problem by using a population, or "swarm", of candidate solutions, referred to as particles. It is a population-based optimization algorithm inspired by the social behavior of birds, which exhibit a type of collective intelligence in their search for food or other resources.

In PSO, each particle represents a candidate solution to the optimization problem, and the position and velocity of each particle are updated iteratively based on the best solution found so far by the particle itself, as well as the

best solution found by the rest of the swarm. The movement of each particle is influenced by its own experience and the experience of the other particles in the swarm, with the goal of eventually converging on the optimal global solution.

Particle Swarm Optimization is performed by initializing a swarm of particles with random velocities and positions (represented as n-dimensional vectors) within the search space. Then, at each iteration, every particle position (representing a solution to the optimization problem) is updated using a pre-defined fitness function. This evaluation is then used as follows:

- If the fitness observed is superior to the fitness of the particle's previously best-known position, it is updated as the particle's new personal best position (**pbest**).

- If the observed fitness is superior to the previous global best fitness, throughout all particles and iterations, it is updated as the swarm's new global best position (**gbest**).

In every iteration, after the evaluation of all particles, the velocity and position of each particle are updated using the following equations:

$$\mathbf{v}(t+1) = w\mathbf{v}(t) + c_1 r_1(\mathbf{pbest} - \mathbf{x}(t)) + c_2 r_2(\mathbf{gbest} - \mathbf{x}(t)) \qquad (2.23)$$

$$\mathbf{x}(t+1) = \mathbf{x}(t) + \mathbf{v}(t+1), \qquad (2.24)$$

where $\mathbf{x}(t)$ and $\mathbf{v}(t)$ are the position and velocity of the particle at iteration t, respectively. w is the inertia weight, c_1 and c_2 are constants that control the influence of the personal best position **pbest** and the global best position **gbest**. r_1 and r_2 are random numbers in the range $[0, 1]$ to introduce stochasticity to the process and avoid getting stuck in local optima. The entire process is illustrated in Figure 2.7.

The key feature of PSO is that it uses the knowledge gained from previous iterations to guide the search for solutions in subsequent iterations. This allows the algorithm to converge quickly to high-quality solutions, even for complex problems with many variables, and in discontinuous problem spaces.

The described algorithm is very similar to the original algorithm proposed by Kennedy and Eberhart [1995], but many variations and hybrids also exist, aiming to improve or adjust the algorithm to solve different types of optimization problems. One such variant is Binary PSO, which can be applied to discrete search spaces.
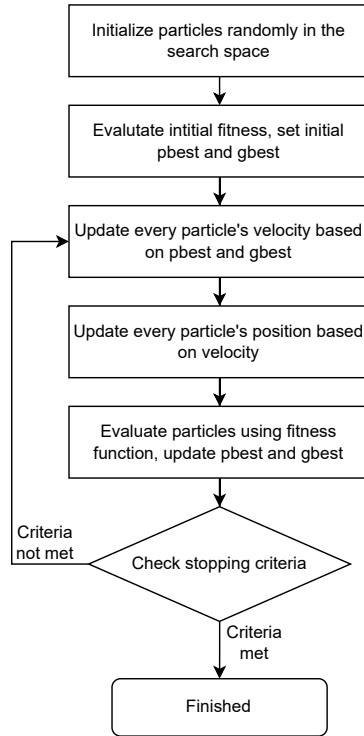
Figure 2.7: Flowchart of Particle Swarm Optimization.

Binary PSO

Binary Particle Swarm Optimization (BPSO), a variation of PSO developed by Kennedy and Eberhart [1997], is specifically designed to solve discrete search spaces, such as those encountered in combinatorial optimization problems. In BPSO, each particle represents a candidate solution as a vector of binary values, in contrast to the continuous-valued vectors employed in traditional PSO.

To function with binary position vectors, the concept of velocity is changed from a physical velocity to a probabilistic one. While the velocity update function remains largely the same, it is applied to the position through a transfer function, which maps the velocity to a probability of each binary position being 0 or 1. Different transfer functions may be utilized, such as sigmoid, S-Shaped, and V-shaped.

In the originally proposed BPSO, the position x_{id} of particle i in dimension d is updated by comparing a random number in the range $[0, 1]$ to a sigmoid

transformation of its velocity v_{id}, according to the following transfer function:

$$x_{id} = \begin{cases} 1 & \text{if } rand(0,1) < S(v_{id}) \\ 0 & \text{otherwise,} \end{cases} \quad (2.25)$$

where S is the sigmoid function defined as:

$$S(v_{id}) = \frac{1}{1 + e^{-v_{id}}}. \quad (2.26)$$

2.5 Clustering

Clustering is an unsupervised learning task that aims to identify patterns in unlabeled data. In simple terms, clustering analysis divides data into groups with common features to keep similar objects together and separate different objects into different clusters.

(a) Hard Clustering   (b) Soft Clustering
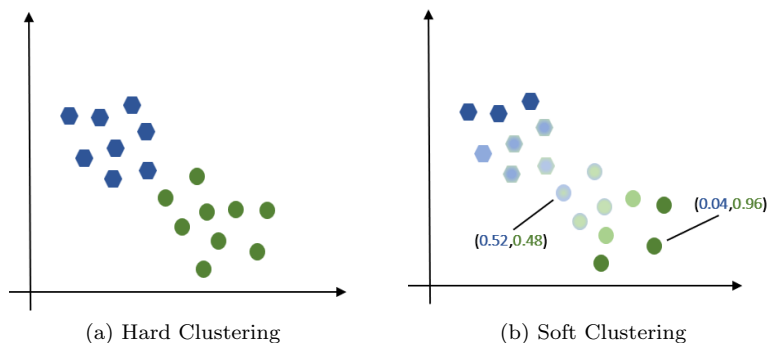
Figure 2.8: Difference between hard and soft clustering. Inspired by Kumar [2022].

There are two distinct approaches to clustering analysis: Hard clustering and soft clustering. In hard clustering, each data point is assigned exclusively to a single cluster based on the similarity of its features, meaning that each object belongs to one, and only one, group. As shown in Figure 2.8, soft clustering, instead of assigning each object to a single group (blue or green), assigns a degree of membership to each data point (shade of blue or shade of green), indicating how well it fits into each cluster.

The following section will introduce a hard clustering approach (k-means clustering) and a soft clustering approach (Fuzzy C-means clustering).

2.5.1   K-means Clustering

The k-means algorithm, introduced by MacQueen [1967], is a simple and straight-forward approach to unsupervised learning for solving the clustering problem. It divides a given data set into a predetermined number of clusters (k clusters). The algorithm starts by selecting k cluster centroids, one for each cluster, and strategically positioning them to maximize their distance from one another. Next, each point in the data set is assigned to the nearest centroid. Once all points have been assigned, the centroids are recalculated as the mean of points in that cluster. This process is repeated until the centroids stop moving or no further changes occur. The ultimate goal of the k-means algorithm is to minimize the objective function defined as:

$$J = \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||^2,$$
(2.27)

where k is the number of cluster, S_i the i-th cluster, and $||x - \mu_i||$ the Euclidean distance between data point x and cluster centroid μ_i.

2.5.2   Fuzzy C-Means Clustering

Fuzzy C-means clustering (FCM), first introduced by Dunn in 1973 [Dunn, 1973], later generalized by Bezdek in 1981 [Bezdek et al., 1984], is a clustering technique that can resemble k-means in a lot of ways. However, as opposed to hard clustering algorithms like k-means, it allows each sample to have a membership degree to multiple clusters. This property of FCM makes it more suitable for dealing with data where the boundaries between clusters may not be well-defined.

The FCM algorithm minimizes the objective function, defined as:

$$J = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m \left\| c_i - x_j \right\|^2,$$
(2.28)

where x_j represents the jth data point, c_i represents the ith cluster center, and u_{ij} represents the membership degree of data point x_j to cluster c_i. N and C are the numbers of data points and clusters. The objective function measures the sum of squared distances between samples and their corresponding cluster centers, weighted by the membership degree. The constraint to ensure the sum of membership degrees for each data point across all clusters is equal to 1 is denoted as:

$$\sum_{i=1}^{C} u_{ij} = 1, \quad \forall j = 1, 2, \ldots, n$$
(2.29)

and the membership degree of the j-th data point in the i-th cluster is defined
as:

$$u_{ij} = \left(\sum_{k=1}^{C} \left(\frac{\|c_i - x_j\|}{\|c_i - x_k\|} \right)^{\frac{2}{m-1}} \right)^{-1}, \tag{2.30}$$

where m is the fuzziness parameter that controls the degree of fuzziness in the
clustering solution. A higher value of m results in a higher degree of overlap
between clusters, and a lower value of m results in a lower degree of overlap.

The FCM algorithm alternates between updating the cluster centers and up-
dating the membership degrees until convergence or a maximum number of iter-
ations is reached. The update rule for cluster centers is defined as:

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{\sum_{j=1}^{C} u_{ij}^m}. \tag{2.31}$$

To minimize Equation 2.28, the *Lagrange multiplier* can be added to the
sequence [Zhang et al., 2017]. The method of *Lagrange multipliers* is an approach
used in mathematical optimization to identify a function's local maximum and
minimum points while considering equation constraints. For multiple constraints,
it can be defined as:

$$\mathcal{L}(x, \boldsymbol{\lambda}) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x), \tag{2.32}$$

where $f(x)$ is the objective function to be optimized, $g_i(x)$ are the constraint
functions, and λ_i are the Lagrange multipliers.

With the incorporation of Lagrange multipliers, the objective function for the
FCM algorithm is revised as follows:

$$J = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \|c_i - x_j\|^2 + \sum_{j=1}^{n} \lambda_j \left(\sum_{i=1}^{C} u_{ij} - 1 \right), \tag{2.33}$$

where λ_j are the Lagrange multipliers introduced to account for the constraint
specified in Equation 2.29. This allows the minimization of the objective function
while considering this constraint.

Chapter 3

State of the Art

In the following chapter, research relevant to this project is presented. The chapter is split into four sections, beginning with an introduction to hyperspectral band selection (BS) in Section 3.1, before discussing relevant BS techniques in Section 3.2. Next, research on different Particle Swarm Optimization approaches relevant to BS is reviewed in Section 3.3. Finally, Section 3.4 covers the essential concepts of PSO-FCM [Zhang et al., 2017] in more detail, for the purpose of extensions presented in subsequent chapters.

3.1   Hyperspectral Band Selection

Technological advancements in imaging spectrometers now allow hyperspectral sensors to capture the spectral reflectance of ground objects on Earth's surface using several narrow bands [Goetz, 2009; Sun and Du, 2019]. The finer spectral resolution of hyperspectral imaging has improved the ability to differentiate ground objects. As a result, the technique has been applied in various practical areas, such as geological mapping [Tan et al., 2020] and ocean monitoring [Muller-Karger et al., 2013], allowing for the analysis and classification of large portions of the Earth's surface.

Although the availability of a large number of spectral bands provides an opportunity for more accurate land cover classification [Patro et al., 2021], HSI processing faces the challenge of the well-known "curse of dimensionality" [Hughes, 1968]. Due to the high dimensionality, the computational burden is huge. To address this, reducing dimensionality and choosing bands that exhibit the highest separability is necessary when working with HSIs [Duan et al., 2021]. Hence, new dimensionality reduction methods and protocols required for HIS processing have become necessary to develop.

Dimensionality reduction methods can be divided into two categories: those based on feature extraction and those based on BS. Early research focused mainly on component analysis for feature extraction. Typical examples include the principal component analysis (PCA) [Harsanyi and Chang, 1994] and independent component analysis [Wang and Chang, 2006], applied to extract a small set of uncorrelated features. However, many criticized feature extraction methods for not preserving the information or physical meaning of the original band values [Petrie et al., 1998; Zhang et al., 1999]. To address this problem, Zhang et al. [1999] proposed a method to separate the hyperspectral bands into similar subspaces and run PCA on each subspace, measuring the similarity of bands using the correlation coefficient metric. Despite this effort, feature extraction methods still reduce the data's usefulness for physical modeling due to the lack of spectral characteristic analysis of the ground objects [Zheng et al., 2017].

Contrastingly, BS, unlike feature extraction, maintains the informational and physical meaning of the original band [Patro et al., 2021]. Due to this advantage, numerous BS methods have been proposed in the academic research of the past decade. However, various challenges remain to be solved in hyperspectral BS.

3.2   Band Selection Techniques

Based on whether labeled training samples are used, BS approaches are classified as supervised or unsupervised. Supervised approaches, such as those discussed in Baisantry et al. [2022] and Su et al. [2014], determine a subset of bands through model training with labeled samples. Even though supervised approaches can deliver high classification accuracy, they rely on prior ground information. In contrast, unsupervised techniques, such as those presented in Zhang et al. [2017] and Zhao et al. [2021], select features without the need for a training set or prior information, relying instead on various statistical metrics and cluster quality assessments. Unsupervised techniques often become preferable for BS due to the scarcity or absence of labeled information [Patro et al., 2021]. This section will primarily focus on unsupervised BS techniques.

3.2.1   Band Quality Measures

Unsupervised quality measures for hyperspectral data are typically based on information-theoretic concepts such as entropy [Zhang et al., 2017], mutual information [Tschannerl et al., 2019], and Kullback-Leibler divergence [Jia et al., 2022]. These techniques do not necessitate manual data labeling. Information-theoretic measures have been explored for selecting bands in HSIs over the past few decades [Conese and Maselli, 1993; Jia et al., 2022], and are often used in conjunction with different search algorithms, such as PSO [Zhang et al., 2017;

Paul and Chaki, 2022], gravitational search [Tschannerl et al., 2019] and NSGA-II [Grimstad, 2019].

There are two main approaches to applying information theory in this context: some research focuses on the entropy of selected bands [Zhang et al., 2017], while others consider redundancy and mutual information between bands [Hossain et al., 2012; Tschannerl et al., 2019]. The entropy approach ensures that only the most representative bands are selected but does not guarantee that the selected bands contain different information. The mutual information approach, on the other hand, does not guarantee high entropy in the selected bands.

Applying the standard mutual information measure for BS in hyperspectral data can lead to suboptimal results, given its incapacity to function effectively as a distance metric. [Hossain et al., 2012; Yang et al., 2017]. To address this issue, Hossain et al. [2012] proposed the use of a measure termed normalized mutual information, which divides the mutual information by the product of the marginal entropies of the two variables to remove the dependency on the entropies of individual variables.

Yang et al. [2017] introduced the use of Disjoint information as a distance measure for comparing spectral bands. This metric computes the joint entropy of two variables and subtracts the mutual information from it. It ensures that variables with substantial individual entropies are prioritized, a vital consideration in BS [Grimstad, 2019].

Further development in the utilization of disjoint information and other divergence measures was explored in a subspace decomposition process by Grimstad [2019]. In this work, the divergence measures were employed to evaluate the differences between adjacent bands, which involved clustering bands in highly correlated subspaces. However, this approach seemed to excessively depend on divergence measures between adjacent bands, potentially leading to the selection of noisy bands due to inevitable discrepancies between adjacent bands. Despite the utilization of disjoint information to help counteract this issue by giving preference to high entropy bands, it did not entirely solve the problem [Grimstad, 2019].

3.2.2 Clustering-based Techniques

Many recent BS approaches incorporate ranking or clustering. Ranking-based techniques [Zhang et al., 2018; Wang et al., 2015] evaluate the quality of each band independently and rank them according to a predetermined metric, such as informativeness or separability. Because the ranking-based techniques mainly focus on scoring criteria for each individual band, the selected bands tend to display a high degree of stability. However, they disregard any correlation information between them. In contrast, clustering-based techniques [Paul and Chaki, 2022;

Yang et al., 2017; Zhang et al., 2017] tend to select bands with lower correlation and less redundant information.

One widely used clustering approach for BS is the k-means clustering algorithm, as employed in Yang et al. [2017] and Paul and Chaki [2022]. Yang et al. [2017] proposes using the k-means algorithm to group bands into clusters based on the disjoint information measure, and shows that k-means clustering can be a simple and efficient approach compared to incremental BS methods. However, due to the intrinsic complexity of spectral bands, the bands can often exhibit overlapping characteristics [Zhang et al., 2017]. Hard clustering, such as k-means, has trouble handling cases that cannot be exactly clustered into a single cluster (see Section 2.5). Soft clustering approaches, such as fuzzy C-means clustering, can therefore be more advantageous for BS as it offers a natural way to handle uncertainty and partial membership. This fuzzy approach to clustering allows for greater flexibility in data classification, as points do not have to be strictly assigned to a single cluster.

Zhang et al. [2017] finds that fuzzy C-means clustering is sensitive to initial conditions and noisy bands, and attempts to solve this challenge by introducing a new unsupervised BS method, PSO-FCM, which employs a PSO-based optimization. Although the PSO hybrid alleviates two limitations of FCM, its sensitivity to initialization and poor computational efficiency, the noise issue is only partially mitigated [Zhang et al., 2017]. Zhang et al. addresses the issue of sensitivity to noisy bands by employing a selection mechanism that selects bands from each cluster with the max entropy criterion. However, the method does not manage to remove noisy bands well enough when the number of clusters is too small.

The selection mechanism in Zhang et al. [2017] can be linked to a more general challenge of clustering techniques: choosing cluster representatives. One of the first clustering-based techniques proposed for BS, Martinez-Uso et al. [2007], solves this by selecting based on the highest average similarity to the other bands in the cluster. In contrast, Yang et al. [2017] selects the representative band for each cluster in a more complex matter using an iterative procedure that maximizes similarity with other bands in the cluster and minimizes similarity with selected bands from other clusters. However, since the selection of representative bands is based only on similarity and not on the bands' information content, noisy bands can affect the algorithms.

Overall, the research suggests there is a risk of producing suboptimal clusters that include noisy bands when using cluster techniques [Patro et al., 2021]. To address this issue, integrating ranking with a clustering-based technique can be beneficial, as shown in Jia et al. [2022]. This research proposes a unified scheme that considers both frequency of occurrence and information entropy as a ranking-based voting strategy for selecting the final band subset after clustering.

3.2.3 Pre-removal of Noisy Bands

The BS process seeks to identify the most informative and distinctive bands while discarding uninformative ones. Notably, bands known to be uninformative can be removed *a priori*, including bands within the water absorption range [Patro et al., 2021]. One approach for removing noisy bands outside the water absorption range is employing a histogram-based method [Kar et al., 2019]. This technique not only identifies uninformative bands but also reduces noise prior to the classification process, which has been shown to improve performance when done prior to target detection [Ji et al., 2019]. Despite the demonstrated effectiveness of this method, the research on automatic pre-removal techniques for noisy bands is limited to the classification problem and target detection. Much of the existing research, including Jia et al. [2022] and Zhao et al. [2021], depends primarily on the manual removal of noisy bands prior to the BS process, often without examining the subsequent impact this has on the effectiveness of the proposed BS techniques.

3.2.4 Superpixel-based Techniques

Superpixel segmentation has been utilized effectively in a broad range of applications, even before they were named in Ren and Malik [2003] [Stutz et al., 2018]. With the increasing quality and interest in remote sensing and HSI data, superpixels have recently gained popularity in the field of HSI analysis [Subudhi et al., 2021]. Their ability to adapt to the spatial structure of objects in the scene makes them a valuable tool for incorporating spatial features in HSI data analysis [Subudhi et al., 2021]. Superpixels can also be used to reduce the number of image primitives for analysis, resulting in a significant decrease in computational complexity [Di et al., 2021]. As a result of these advantages, superpixels have been widely utilized in various areas of HSI processing, such as classification [Wang et al., 2020], spectral unmixing [Wang et al., 2017], anomaly detection [Ren et al., 2019], image restoration [Fan and Huang, 2021], and denoising [Sun et al., 2018]. However, most research has focused on utilizing superpixels directly in the classification of HSIs rather than in BS. Only a limited number of recent research have explored the integration of superpixels in BS [Zhao et al., 2021; Yang et al., 2018; Jia et al., 2022].

Various superpixel algorithms have been successfully utilized in the HSI context, mainly divided into two categories: graph-based methods such as ERS [Yang et al., 2018; Zhao et al., 2021], and gradient-based methods such as SLIC [Jia et al., 2022]. ERS provides segmented superpixels with shape adaptability resulting in good boundary adhesion, whereas SLIC enforces connectivity from the start, resulting in good compactness and regularity [Jia et al., 2022]. Hence, Jia et al. [2022] implements multiple superpixel segmentation approaches to generate maps at various scales, allowing for the complementarity of multiple superpixels.

However, this comes with a trade-off in terms of computational efficiency. It is also worth noting that these algorithms were not initially developed for HSIs, and require adjustments to function properly for BS [Subudhi et al., 2021].

Generally, when superpixel information is utilized in BS, the variability between superpixels and within a single superpixel is considered. Zhao et al. [2021] proposed an unsupervised BS approach termed Spectral-Spatial GA (SSGA), which employs a GA with a fitness function that measures the performance of the candidate subset of bands using a superpixel-based unsupervised trace ratio criterion with spectral-spatial capabilities. In contrast, Yang et al. [2018] utilizes spatial information from superpixels defining two new band criteria, one based on Metric Learning and one based on Representation Learning. Recently, Jia et al. [2022] proposed a superpixel-level Kullback-Leibler distance of bordering bands to divide bands into clusters, encouraging within-cluster correlation and lower correlation across clusters. The proposed method subsequently employs a second superpixel-level distance measure, to select representative bands from each cluster. On the one hand, these fitness functions/distance measures offer the advantage of being able to more effectively evaluate the variability both within and between superpixels and exploit both spectral and spatial information. On the other hand, it also presents the challenge of an increase in computational complexity.

Although the integration of spatial information in band selection through superpixels presents promising potential, it remains a research area in need of further investigation.

3.3   Particle Swarm Optimization

Particle swarm optimization (PSO), introduced in 1995 by Kennedy and Eberhart [Kennedy and Eberhart, 1995], has gained significant attention from researchers within the HSI field. The following subsections discuss some influential contributions that apply PSO for BS.

3.3.1   Discrete PSO

In PSO, particles must typically navigate a continuous search space to find the optimal solution. However, in feature selection problems like BS, the use of continuous values is often not advantageous. This problem is inherently a discrete optimization problem, and the PSO algorithm often needs modifications or direct manipulation to solve discrete problems effectively.

One approach applied to BS is Binary PSO (BPSO) [Zhang et al., 2017]. The performance of BPSO, first introduced in Kennedy and Eberhart [1997], highly depends on transfer function selection [Mirjalili and Lewis, 2013]. As such,

various transfer functions, including sigmoid, S-shaped, and V-shaped, have been proposed, as discussed in Shami et al. [2022].

V-shaped transfer functions have demonstrated superior performance in terms of the convergence rate and escaping from local minima, compared to S-shaped transfer functions [Mirjalili and Lewis, 2013]. However, V-shaped functions have the potential to get stuck in local optima due to their reliance on the velocity of the PSO method. With the V-shaped transfer functions, if the current position is a local optimum and the velocity approaches zero, the new position will remain the same as the current position [Beheshti, 2019]. To mitigate this problem, Beheshti [2019] introduced a mirrored transfer function that varies over time. Its effectiveness has been tested on high dimensional 0–1 knapsack problems, and the results indicate that the proposed transfer function performs better than the S-Shaped and V-shaped transfer functions. However, this mirrored transfer function's performance has not been examined on feature selection problems [Shami et al., 2022].

Another approach to PSO in a discrete space involves direct modification of the search space, as exemplified by the Multi-Objective Discrete Particle Swarm Optimization (MODPSO) [Gong et al., 2014]. In MODPSO, particle position and velocity are discrete, achieved by modifying the velocity and position update function. This approach simplifies the transition from a continuous to a discrete search space. However, the MODPSO algorithm incorporates a predefined probability, p, for selecting random particles to update velocity. This can impact exploration and exploitation, potentially resulting in the loss of promising solutions when particles are chosen randomly from the nondominated relationship during the update process [Shami et al., 2022].

3.3.2   PSO for Band Selection

Many BS techniques in hyperspectral imaging pose challenges regarding computational complexity and time requirements due to exhaustive feature searches. Therefore, PSO has gained recognition in hyperspectral BS, owing to its ability to navigate large solution spaces effectively and converge on optimal solutions [Sun and Du, 2019].

One of the earlier efforts in using PSO for BS was presented by Su et al. [2014]. They developed a two-level PSO algorithm (2PSO) that determines the optimal combination of bands and the number of bands to be selected in a supervised manner. However, the results revealed the method to be computationally inefficient.

For unsupervised BS, a hybrid approach combining PSO with a genetic algorithm (GA) has been proposed [Ghamisi and Benediktsson, 2015]. The genetic algorithm generates initial particle positions and velocities, and the PSO

algorithm then searches for the optimal solution. This hybridization leverages the global search capabilities of GA and the fine-tuning abilities of PSO. Later, Zhang et al. presented an unsupervised BS approach (PSO-FCM) that combines fuzzy clustering with PSO [Zhang et al., 2017]. The PSO-FCM model is further explained in the following section.

Prior research suggests that the characteristics of PSO, such as its efficiency and insensitivity to initialization, make it a well-suited choice for BS. However, most prior research considers only the HSI's spectral properties, ignoring spatial properties [Paul and Chaki, 2022]. To address this, Paul and Chaki [2022] recently proposed incorporating spatial (gradient) information within the objective function of PSO. The proposed method requires an initial step applying the k-means algorithm to mitigate spatial redundancy before applying the PSO algorithm for BS, introducing computational inefficiency to the overall process.

3.3.3   Fuzzy Clustering with PSO

FCM clustering is commonly used to divide a dataset into a collection of fuzzy groups and has become a popular technique because of its simplicity and unsupervised approach [Dhanachandra and Chanu, 2020]. However, the FCM algorithm is highly influenced by its initial conditions and can be disrupted by noisy data, making it prone to getting stuck in locally optimal solutions [Zhang et al., 2017]. Therefore several hybrid methods have been proposed to address the limitations inherent in FCM, and to leverage the superior clustering outcomes offered by PSO.

In Zhang et al. [2017], PSO helped overcome FCM sensitivity to initialization and the tendency to get stuck in local optima when applied to BS. The PSO algorithm proposed is applied to optimize FCM, replacing the Lagrange multiplier method, often employed with FCM (as explained in Section 2.5). On the other hand, the hybrid showed to be highly affected by noise when the number of the selected bands was small.

In order to address the issue of noise, Dhanachandra and Chanu [2020] proposed an image segmentation technique that integrates Dynamic Particle Swarm Optimization (DPSO), FCM, and a noise reduction method leveraging spatial neighborhood information. Recently, Verma et al. [2021] proposed another hybrid algorithm (FCM-PSO) that mitigates local minima trapping issues. This is achieved by utilizing PSO to compute the centroid of its cluster through a candidate solution.

Overall, research has shown that PSO's global optimization searching capabilities can address the limitations of FCM, leading to improved clustering results [Dhanachandra and Chanu, 2020; Verma et al., 2021; Zhang et al., 2017]. While recent research has explored the application of FCM combined with PSO in im-

age processing [Dhanachandra and Chanu, 2020; Verma et al., 2021], further investigations are warranted to explore its potential in the domain of BS.

3.4 PSO-FCM

The unsupervised BS model, PSO-FCM, which combines a binary PSO with fuzzy C-means clustering, is proposed in Zhang et al. [2017]. Since this research intends to extend this hybrid approach, the present section aims to explain the essential concepts of PSO-FCM, specifically focusing on the formulation of the objective function and the process of cluster representative selection, which remain the same for the model proposed in this work.

3.4.1 Objective Function

The objective function employed in PSO-FCM is a slightly modified version of the standard FCM objective function as presented in Section 2.5.2. For each particle in PSO-FCM, representing a candidate solution of cluster centers, the objective function J is calculated as follows:

$$min \ J = \sum_{i=1}^{C} J_i = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m d_{ij}^2, \tag{3.1}$$

with C being the number of cluster centers, N the total number of bands, m the fuzzy exponent, d_{ij} the Euclidean distance between band i and j. Finally, u_{ij} is the fuzzy membership of band j in cluster i, calculated as:

$$u_{ij} = \frac{\sum_{k=1}^{C} d_{kj}^{\frac{2}{m-1}}}{d_{ij}^{\frac{2}{m-1}}}. \tag{3.2}$$

3.4.2 Cluster Representative Selection

The cluster representative selection method proposed in PSO-FCM is one based on entropy. For each cluster, the band with the max entropy is selected. The entropy of band Y is defined as follows:

$$H(Y) = -\sum_{y \in \Omega} p(y) \log_2 p(y), \tag{3.3}$$

where $p(y)$ denotes the grayscale histogram probability distribution of band Y.

Chapter 4

Model Architecture

This section outlines the architecture of the proposed BS model, Superpixel-enhanced PSO-FCM (SPPF). The SPPF model operates in three distinct steps, each contributing to the overall objective of identifying and selecting the most relevant bands for classification. Figure 4.1 provides a visual representation of the SPPF model architecture. In the first step, the image is segmented into superpixels that share similar characteristics. The second step of the SPPF model involves clustering bands with similar attributes. The clustering process is performed by a PSO-FCM [Zhang et al., 2017] hybrid that is modified to incorporate the superpixel map generated in Step 1. Finally, in the third step, a single band is selected from each cluster to serve as a representative for that particular cluster. The bands produced through this step form the output of the SPPF BS model.

4.1   Superpixel Segmentation

In the initial step of the SPPF model, the HSI is segmented into superpixels, to be utilized in the subsequent BS process. A modified version of the Simple Linear Iterative Clustering (SLIC) method is chosen for this task due to its excellent performance and efficiency. This efficiency is crucial because integrating superpixels is intended to increase the efficiency of the BS process. Moreover, improving efficiency in subsequent image-processing tasks is a primary motivation for BS, making efficiency a vital factor in design choices. In addition, SLIC offers flexibility in terms of controlling the number of superpixels and the compactness of the generated superpixels, allowing us to adapt the segmentation process according to the specific dataset.
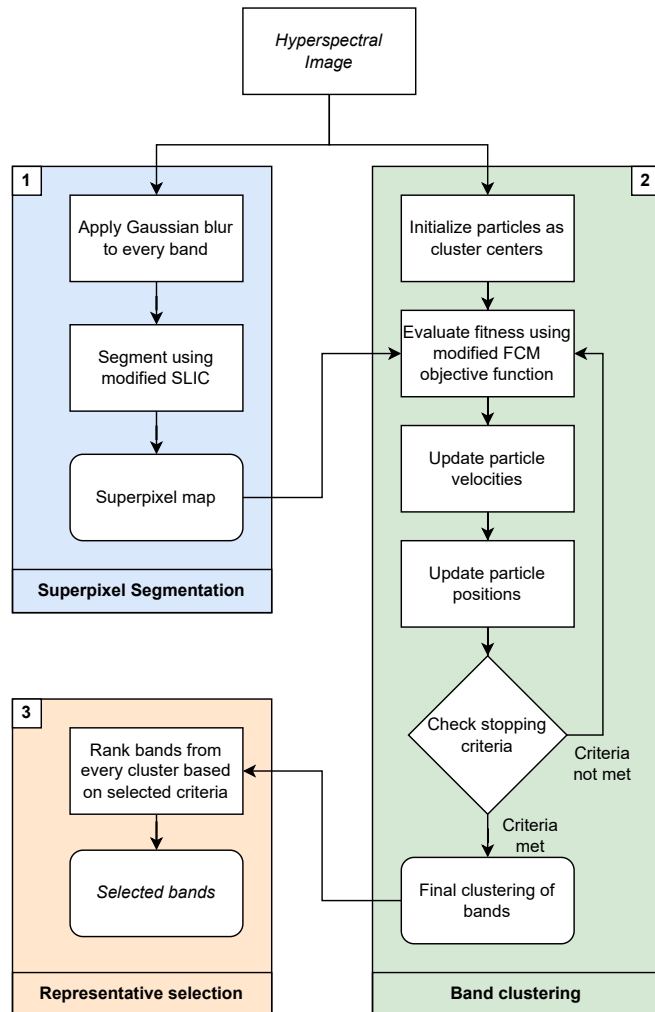
Figure 4.1: SPPF Model Architecture.

4.1.1 Preprocessing: Gaussian Blur

Before segmenting the HSI using the modified SLIC algorithm, a slight Gaussian blur is applied to every band. This preprocessing step is done to reduce high-frequency noise present in the image, which can cause the superpixel segmentation to produce irregular and fragmented regions. By smoothing the image with a Gaussian blur, the segmentation process can better capture the underlying structures and more accurately outline homogeneous regions within the image.

The Gaussian blur is applied by convolving each band with a Gaussian filter. The Gaussian filter, denoted by $G_\sigma(x, y)$, is defined as:

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{4.1}$$

where σ is the standard deviation of the Gaussian distribution, which controls the amount of blur, and (x, y) are the pixel coordinates. The filtered band, denoted by B_{blur}, is obtained by convolving the original band B with the Gaussian filter:

$$B_{\text{blur}}(x, y) = B(x, y) * G_\sigma(x, y), \tag{4.2}$$

where $*$ denotes the convolution operation. Figure 4.2 shows an example of a resulting band after applying Gaussian blur, illustrated as a grayscale image.

(a) Original grayscale image   (b) After applying Gaussian blur
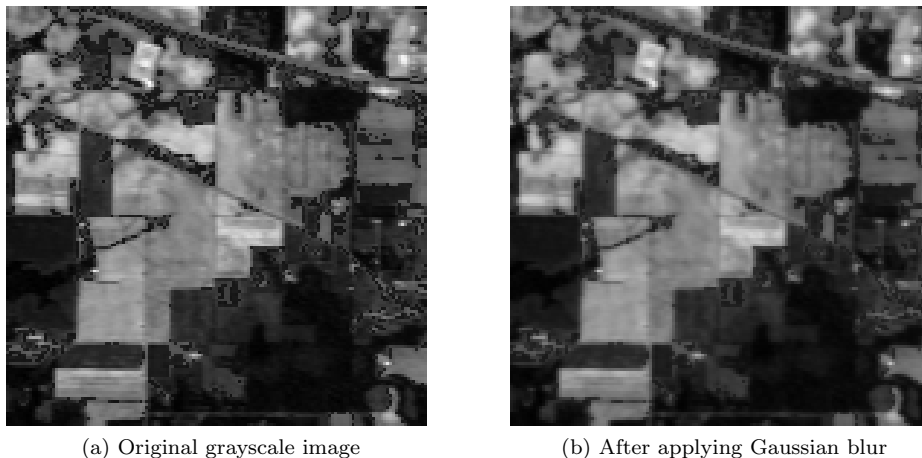
Figure 4.2: Grayscale image of band 30 of the Indian Pines dataset after normalizing spectral intensity values to [0, 255] and the same band after applying Gaussian blur.

4.1.2 Modified Distance Measure for SLIC

The original SLIC algorithm compares pixels in the CIELAB color space, and does not support HSIs. Because of this constraint , the superpixel segmentation in SPPF instead operates on the intensity value in each spectral band to calculate the spectral distance as an alternative to the color distance metric employed in the original SLIC algorithm. The distance in spectral intensity between pixels i and j in a particular band image is defined as:

$$d_{si}(i,j) = \sum_{b} (x(i) - x(j))^2,$$ (4.3)

where b is the total number of bands in the HSI, and $x(i)$ is the intensity value of pixel i in that particular band. With this adjustment, the distance metric used in the superpixel segmentation of SPPF is subsequently defined as:

$$d(i,j) = d_{si}(i,j) + w_{sp}d_{sp}(i,j),$$ (4.4)

where w_{sp} is the spatial weight, and $d_{sp}(i,j)$ is the euclidean distance in the spatial dimensions. This modified distance metric allows the segmentation process to consider all spectral bands to compute a good clustering.

4.1.3 Parameter Selection

Due to the modifications in the distance measure of the SLIC algorithm, the standard recommended values for hyperparameters no longer yield optimal results. Consequently, preliminary tests were conducted to identify new recommended values for the superpixel segmentation step. The recommended hyperparameters for the superpixel segmentation phase of SPPF are presented in Table 4.1. These values have been found to provide good segmentation quality and computational efficiency in general. Instead of specifying a recommended number of superpixels, a recommended average size of superpixel is specified, since the number of required superpixels largely depends on the spatial dimension of the image. Figure 4.3 shows the importance of choosing a high enough number of superpixels. It can be seen that when only 100 superpixels are constructed, entire features such as the fields in the center of the site, are covered in a single, or just a few, superpixels.

4.2 Band Clustering

The second step in the SPPF model's architecture is to cluster the image's bands into the same number of clusters as the desired number of bands to select. By grouping bands with similar hyperspectral characteristics, a single band can later

Hyperparameters Superpixel Segmentation	
σ	0.5
Superpixel size	8x8 - 12x12
w_{sp}	1,000 - 10,000

Table 4.1: Recommended hyperparameters for the superpixel segmentation step of SPPF.

(a) 100 superpixels (b) 200 superpixels (c) 300 superpixels
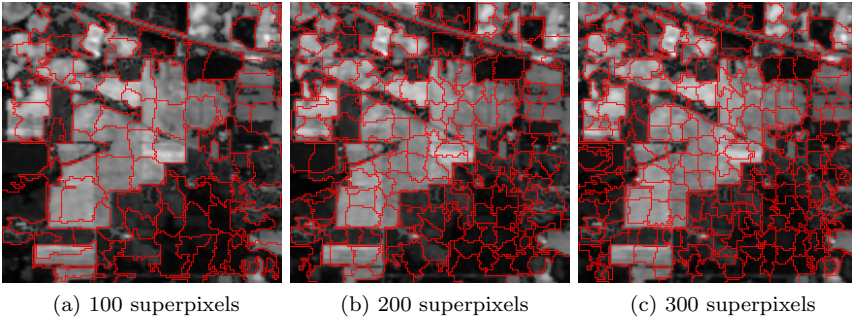
Figure 4.3: Indian Pines image segmented into different number of superpixels.

be selected from each cluster to represent it. This aims to select bands with high informational value, with little redundancy between them. In theory, bands that are clustered together should have a significant overlap of information.

SPPF employs a superpixel-enhanced PSO-FCM algorithm for the clustering process. This hybrid approach leverages the strengths of both PSO and FCM to produce high-quality clustering results, while also benefiting from the spatial information gathered from the segmentation step.

4.2.1 Particle Swarm Optimization

In the context of SPPF, PSO is used to search for the optimal clustering of bands. PSO is chosen for its ability to explore the search space efficiently and effectively. This exploration enables the identification of optimal cluster centroids, allowing for a better selection of representative bands with distinct hyperspectral characteristics. Furthermore, PSO's flexibility and adaptability allow it to be easily combined with other algorithms, such as FCM, and to incorporate superpixel information. The proposed PSO variant, similarly to the one proposed in PSO-FCM [Zhang et al., 2017], is specifically designed to address the problem

of selecting n spectral bands from a HSI to be used as cluster centroids in the FCM objective function. However, unlike PSO-FCM, the SPPF model employs a continuous search space. Its primary advantage over a typical combination-based PSO variant comes from its ability to exploit the neighboring band's spectral similarities, allowing for more directed exploitation and potentially providing faster convergence.

Particle Representation

Similar to PSO-FCM [Zhang et al., 2017], each particle in SPPF represents a candidate solution for selecting cluster centroids. For every iteration, each particle is decoded using a mapping function and evaluated using the objective function from FCM. The i-th particle's position is represented as a continuous c-dimensional vector $X_i = \{x_{i1}, x_{i2}, ..., x_{ic}\}$, where c is the number of clusters . The value of the particle in each dimension represents a chosen cluster centroid.

Since each dimension represents the index of a chosen cluster centroid, all dimensions of the search space have a lower bound of 0 and an upper bound equal to the number of total bands available minus one. This is illustrated in Figure 4.4, where the dimensionality of two indicates that two bands are to be selected, and the upper bounds for each dimension indicate that there are 11 possible bands to select from. At initialization, bands are selected randomly and set as initial cluster centroids to generate initial particles. This ensures a diverse set of initial solutions which allows the swarm to explore various regions of the search space and increase the likelihood of finding an optimal selection of bands.

Decoding of Solutions

The particles reside in a continuous space and must be decoded by the mapping function to determine which bands will be selected as cluster centroids for evaluation. Contrary to PSO-FCM's strategy of adopting a probability transition (as mentioned in Section 3.4), SPPF employs a rounding-off method. This choice is driven by its simplicity and computational efficiency. The mapping function converts the continuous values from the search space into discrete band indexes to be evaluated as cluster centroids (CC) using the objective function from FCM. This mapping bridges the gap between the continuous nature of the PSO algorithm and the discrete nature of the problem. The mapping function is defined as:

$$CC_i = round(X_i), \tag{4.5}$$

where $round(X_i)$ is the resulting vector from rounding every dimension of X_i separately to its closest integer value.
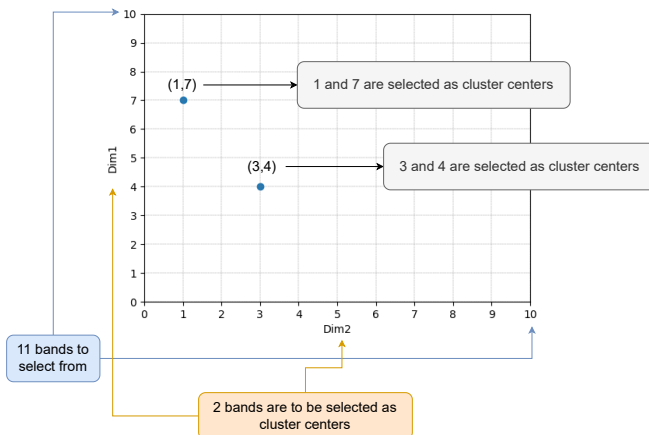
Figure 4.4: PSO Representation showing a case where two of eleven possible bands are to be selected with a PSO population size of two.

Particle Movement

The particles move through the search space by adjusting their positions and velocities according to their personal best and global best solutions. Additionally, a maximum velocity v_{max} is introduced as a fraction of the range between the upper and lower limits of the search space. This parameter plays a crucial role in controlling the algorithm's convergence and maintaining stability during the optimization process. The value of the fraction is decided during preliminary testing.

Repair Mechanism

The FCM objective function inherently discourages selecting duplicate bands as cluster centroids. However, this problem might still occur when the number of selected bands is high. This violates the problem constraint of selecting exactly n unique bands. To prevent this, SPPF employs a repair mechanism that detects and corrects these violations by replacing duplicate selected bands with randomly selected new band indices.

Termination

The PSO algorithm terminates once it reaches the pre-defined maximum number of generations. This condition acts as an upper limit on the number of iterations

the algorithm performs. The model also checks if there has been no improvement in the global best solution for a specified number of iterations, denoted as t. If the global best position remains unchanged for t consecutive iterations, the algorithm assumes that it has reached a point where further exploration is unlikely to yield significant improvements and terminates the optimization process. The value for t should be chosen in consideration with all other parameters that control the speed of convergence, such as the inertia weight w and the cognitive and social coefficients c_1 and c_2.

4.2.2   Objective Function

Similar to PSO-FCM [Zhang et al., 2017], SPPF employs the FCM-based objective function J as the objective function for the PSO algorithm (as detailed in Section 3.4). FCM offers a natural way to handle uncertainty and partial membership, which is particularly relevant for the BS model, where spectral bands can often exhibit overlapping characteristics (see Section 3.2.2). FCM is less sensitive to noise and outliers compared to hard clustering algorithms like k-means. This is because FCM can assign partial memberships to noisy data points, reducing their impact on the overall clustering. HSIs often suffer from noise and inconsistencies in the data, and FCM's inherent ability to handle noise makes it a suitable choice.

SPPF calculates the fuzzy membership and value of the objective function J for each candidate solution evaluated during clustering with PSO. The objective function J is to be minimized. Fuzzy membership u_{ij} and value of the objective function J is defined as:

$$u_{ij} = \frac{\sum_{k=1}^{C} d_{kj}^{\frac{2}{m-1}}}{d_{ij}^{\frac{2}{m-1}}} \tag{4.6}$$

$$J = \sum_{i=1}^{C} J_i = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m d_{ij}^2, \tag{4.7}$$

where N is the number of total bands. C is the number of clusters. u_{ij} denotes the fuzzy membership of the j-th band corresponding to the i-th cluster, and the parameter m is the fuzzy exponent which decides how hard the clustering should be. d_{ij} is defined as the distance between the i-th cluster centroid and the j-th band. Section 4.2.3 delves deeper into the different distance metrics employed in SPPF.

4.2.3 Distance Metrics

To better capture both spectral and spatial information in the band clustering process, the distance metric used in the FCM objective function is modified to incorporate superpixel information derived from the segmentation step. This allows the SPPF algorithm to consider the local spatial context provided by the superpixel map when clustering the bands. The goal of this is to increase the quality of the resulting clusterings, and increase the efficiency of the search process by reducing the size of data used for distance calculations in FCM.

The following distance metrics are proposed:

Superpixel Euclidean Distance

The first proposed metric, labeled the Superpixel euclidean distance, adapts the distance metric used in the original PSO-FCM BS algorithm [Zhang et al., 2017], modified to take advantage of the superpixel information. The Superpixel Euclidean distance considers the Euclidean distance between the aggregated values (mean) of the superpixels, rather than individual pixels. Using aggregated values from each superpixel aims to encourage the clustering of bands that have similarities in both spatial and spectral attributes. In addition to incorporating spatial and spectral information, the Superpixel Euclidean distance reduces the required distance calculations, as it operates on aggregated superpixel values rather than individual pixels. This reduction in computational complexity aims for faster clustering at the cost of higher overhead complexity, as the superpixels have to be computed and aggregated prior to BS.

The superpixel Euclidean distance between band i and j is defined as:

$$d_{i,j} = \sqrt{\sum_{k=0}^{sp} (x_i(k) - x_j(k))^2}, \tag{4.8}$$

where sp is the number of superpixels and $x_i(k)$ is the mean value of superpixel k in band number i. The difference between the superpixel Euclidean distance and the normal Euclidean distance is highlighted in Figure 4.5. It shows how the Superpixel Euclidean distance operates on aggregated values instead of individual pixels.

Superpixel KL-distance

The second proposed metric is a KL-Divergence-based measure, the Superpixel KL distance. Similar to the Superpixel Euclidean distance, this metric is designed to take advantage of the superpixel information using the superpixel's

(a) Euclidean distance   (b) Superpixel Euclidean distance
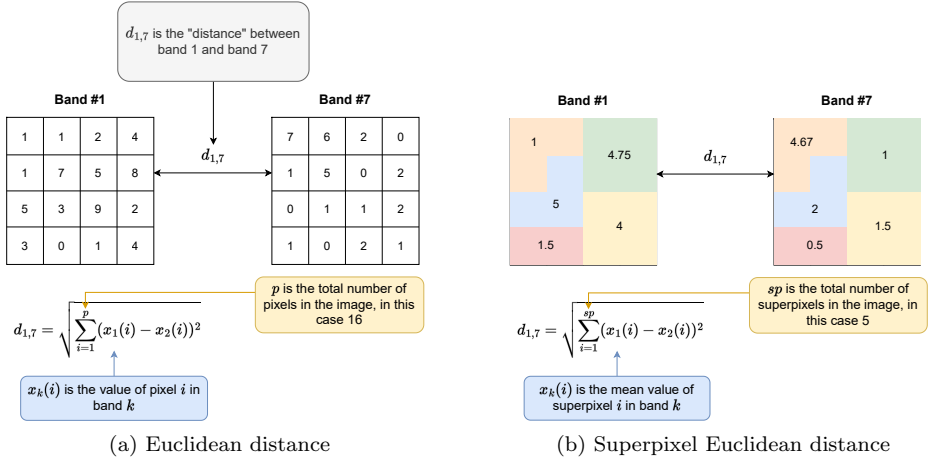
Figure 4.5: Euclidean distance and Superpixel Euclidean distance. The colored regions in (b) signify different superpixels; the number within them is their aggregated values in that band.

mean values. As an information-based distance measure, it possesses an inherent advantage over direct distance measures such as Euclidean distance, owing to its ability to capture the information content of the data being compared. Given that the KL-Divergence quantifies the difference in entropies between spectral bands, this approach is expected to promote the segregation of bands with distinct entropy profiles into separate clusters.

As described in Section 2.2, for KL-Divergence to be applied as a distance metric, it is symmetrized. Furthermore, normalized grayscale histograms are calculated to obtain the probability distributions of each band. The grayscale histogram for a band, i, is calculated by binning all k superpixels, represented by its mean value $x_i(k)$, to N bins. The resulting probability distributions are then used to calculate the Superpixel KL distance between band i and band j as:

$$d_{i,j}^{KL} = \sum_{n=1}^{N} p_{n,i} \log \frac{p_{n,i}}{p_{n,j}} + p_{n,j} \log \frac{p_{n,j}}{p_{n,i}}, \qquad (4.9)$$

where N is the number of bins and $p_{n,i}$ and $p_{n,j}$ are the values in the n-th bin of the grayscale normalized histograms of the i-th and j-th band.

Using the aggregated value of each superpixel instead of each pixel reduces the number of values used to compute the histogram. However, this leads to a more concentrated distribution over fewer bins, as shown in Figure 4.6. The figure demonstrates the notable dissimilarity between the distributions of gray levels,

with a higher degree of disparity observed for fewer superpixels (b). The reason for this can be the segmentation of superpixels in varying sizes and numbers of pixels, leading to situations where a mean superpixel value representing a larger pixel count has a similar impact as a superpixel-mean value corresponding to a smaller pixel count, thereby resulting in divergent distributions.

(a) All Pixels   (b) 100 Superpixels
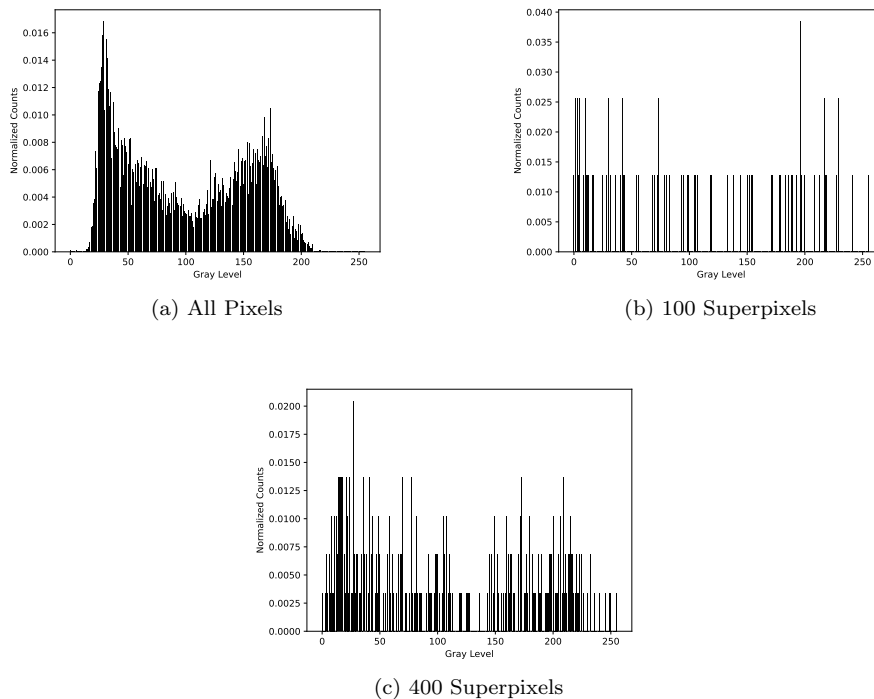
(c) 400 Superpixels

Figure 4.6: Grayscale normalized histograms of band 111 from Indian Pines dataset, using all pixels (a), mean-values from 100 superpixels (b), and mean-values from 200 superpixels (c).

Superpixel Disjoint Information

Disjoint information is the third distance metric proposed using the superpixel's mean values. The Superpixel Disjoint Information adapts the disjoint information distance proposed in Yang et al. [2017], modified to take advantage of the superpixel information similar to the Superpixel KL distance. Disjoint information is

a value derived from the joint entropy and mutual information, as described in Section 2.2, which characterizes the information in band i and band j that is not shared between them. Therefore, it is comparable to the KL divergence in that it describes a disparity rather than a resemblance. However, unlike KL divergence and mutual information, disjoint information is already symmetric and satisfies all criteria required to be considered a metric.

The disjoint information between two bands i, and j, is determined by the following calculation:

$$d_{i,j}^{DI} = \sum_{x_i^n} \sum_{y_j^m} p(x_i^n, y_j^m) log \frac{p(x_i^n)p(y_j^m)}{p(x_i^n, y_j^m)}, \tag{4.10}$$

where $p(x_i^n)$ is the n-th bin of the grayscale normalized histograms of the i-th band, while $p(y_j^m)$ is the m-th bin of the histograms of the j-th band. The grayscale normalized histograms are calculated similarly to the ones described in Section 4.2.3. Finally, $p(x_i^n, y_j^m)$ denotes the corresponding joint probability distribution.

Applying disjoint information instead can help to address the issue of a concentrated distribution over fewer bins because it is less sensitive to differences between small probabilities, which dominate the KL-Divergence calculation. Nevertheless, this will be at the cost of higher overhead complexity, since the joint probability distributions also need to be calculated prior to BS.

Superpixel-level KL Divergence-L1Norm

The final metric proposed is labeled Superpixel-level KL Divergence-L1Norm, and is adopted from Jia et al. [2022], with slight modifications to fit SPPF.

The proposed distance metric computes KL divergences between bands at the superpixel level. L1-norm is applied to generate the KL-L1 distance matrix, enabling the incorporation of spatial information within the homogeneous regions of the superpixels.

The computing process is shown in Figure 4.7. The approach involves first considering each homogeneous region of the superpixel map as a single entity and then calculating the KL divergence distance between all bands within each superpixel. This operation produces a distance matrix $(B \times B)$ for each superpixel where B represents the number of bands, as shown in Figure 4.7 where two superpixels are highlighted in green and yellow color, as well as their resulting distance matrix. Note that number of bands B turns 90 degrees between the first and the second part. The individual distance matrices for each superpixel are then stacked together to form a 3D distance tensor of size $B \times B \times N$, where N represents the number of superpixels in the image. Finally, the L1 norm of the 3D distance cube (highlighted in red) is computed to obtain the distance matrix

$(B \times B)$ between the entire image bands. This KL-L1 distance matrix can be computed with the following formula:

$$d_{i,j}^{KL-l1} = \sum_{n=1}^{N} \sum_{w=1}^{W_n} |p_{n,w,i} \log \frac{p_{n,w,i}}{p_{n,w,j}}|, \qquad (4.11)$$

where $d_{i,j}^{KL-l1}$ represents the KL-L1 norm distance measures between the i-th and j-th bands of the n-th superpixel. The variables $p_{n,w,i}$ and $p_{n,w,j}$ refer to the histogram statistics of the w-th pixel in the n-th superpixel for the i-th and j-th bands.

Notably, the proposed distance metric entails a certain degree of overhead complexity, since it involves calculating the distance within and between each superpixel. Nonetheless, the benefits of incorporating inner and outer variability in the metric can outweigh this disadvantage. It enables a more comprehensive characterization of the superpixel structure, which can be important when classifying.
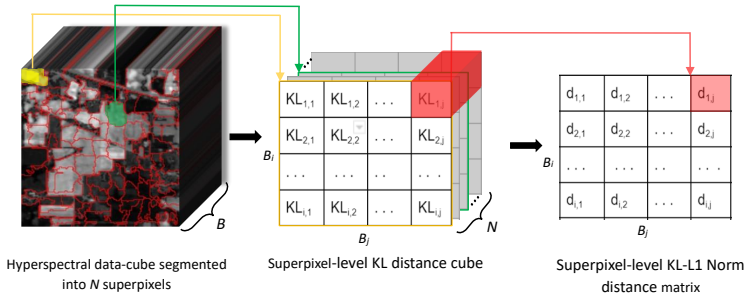


Figure 4.7: Computation process of superpixel-level KL-L1 Norm distances. Inspired by Jia et al. [2022].

4.3   Choosing Cluster Representatives

Once clustering is performed using the SPPF algorithm, a single band is selected from each cluster as the representative. This step is crucial for creating a subset of bands that effectively represents the entire HSI, while minimizing redundancy.

The following ways of selecting cluster representatives are proposed:

Highest Entropy

One approach for the selection of representative bands is through the utilization of entropy as a selection criterion, as demonstrated in the PSO-FCM model [Zhang et al., 2017]. Entropy serves as a measure of randomness or uncertainty present in the spectral band. By selecting the band with the highest entropy within each cluster, the intention is to maximize the information content of the chosen bands. High entropy indicates increased variability and decreased predictability in the data, rendering it more informative. Consequently, this method reduces the likelihood of selecting inherently noisy bands.

A limitation of the entropy-based method is the disregard of the similarity between the representative band and the remaining bands within the same cluster. This may result in the selection of redundant bands, containing overlapping or highly correlated information.

Central Tendency

The second method for selecting representative bands from clusters involves identifying the band closest to the mean of the cluster. In this approach, the objective is to find a band that best represents the central tendency of the spectral bands within the cluster, providing a higher likelihood of being representative of a majority of the bands in the cluster.

However, a potential disadvantage of this approach is that the band closest to the mean may not necessarily contain as much information as other bands in the cluster. Consequently, the selected band may only partially capture the range of information present within the cluster, leading to a suboptimal representation.

To identify the band closest to the mean of a cluster, the arithmetic mean of all bands within that cluster is calculated for each pixel value. Then each pixel's spectral intensity is compared with the computed mean for every band, and the band with the lowest overall difference from the mean is selected.

Hybrid Weighted-sum Criterion

A hybrid approach that combines the highest entropy and central tendency methods is designed to balance the benefits of both methods while addressing their individual limitations. This approach aims to select a representative band that both maximizes information content and closely resembles other bands in the cluster.

To compute the weighted sum, it is necessary to first compute and standardize the entropy and central tendency criterion to the same range. The selection

criterion is then defined as follows:

$$HWS(B) = w_e SE(B) - w_{ct} SCT(B), \qquad (4.12)$$

where w_e and w_{ct} are the weights of the entropy criterion and central tendency criterion respectively. $SE(B)$ is the standardized entropy of band B, and $SCT(B)$ is the standardized distance to the mean band. The second term is subtracted instead of added due to the central tendency criterion being a minimization criterion. The weights should be adjusted depending on the desired balance between maximizing information content and ensuring representation of the cluster's central tendency.

Hybrid Ranking Criterion

The hybrid ranking criterion, proposed in SPPF for selecting a representative band, combines the entropy and central tendency criteria to achieve a balance between high information content and close resemblance to the central tendency of other bands in the cluster. In this approach, each band is ranked based on its entropy and distance to the mean, which serves as the chosen central tendency measure. The overall rank is computed by summing the entropy-based rank and the distance-to-the-mean-based rank. The representative band is then determined as the one with the lowest overall rank, providing a good balance between information content and representativenes. If there is a tie, the band with the highest entropy is selected.

4.4 Model Evaluation

To assess the quality of SPPF, an SVM classifier that utilizes only the selected bands from the algorithm is employed. The performance of the classification algorithm will be evaluated using two metrics. The first is overall accuracy (OA), simply the number of correctly labeled test samples divided by the total number of test samples. The second metric is average accuracy (AA), the average number of correctly labeled test samples of each class. This metric can provide insight into how the selected bands perform in classifying different classes. Both the mean and standard deviation of these values will be recorded across multiple independent classification runs. Additionally, the runtime of the SPPF model is considered an important metric for comparison.

The SVM is selected as the classification algorithm because it performs well with small training sets, has an available high-quality implementation, and has reasonable computational requirements. More evaluation details are presented in Chapter 5.

Chapter 5

Experiments and Results

This chapter introduces the experiments conducted to answer the research questions of this thesis. Preliminary testing is presented in Section 5.1, and an introduction to the main experiments and what knowledge they seek to find is presented in Section 5.2. The experimental setup is presented in Section 5.3, and finally the experiment results in Section 5.4.

5.1 Preliminary Testing

The preliminary tests determine suitable hyperparameters for the different components of SPPF. The intent is not to find the optimal parameters for every variation of different design decisions but rather to identify a singular set of parameters that perform well and consistently across all datasets and combinations of design decisions. These parameters are then utilized throughout the subsequent experiments of this study.

For the superpixel segmentation algorithm, preliminary experiments first determined the optimal number of superpixels for each hyperspectral dataset, as shown in Figure 5.1. Subsequently, the spatial weight w_{sp}, and the standard deviation for the Gaussian blur σ, were determined through a combination of quantitative and qualitative analysis. The resulting hyperparameters for the superpixel segmentation step of SPPF can be seen in Table 5.1.

(a) Indian Pines   (b) Salinas   (c) Pavia Center
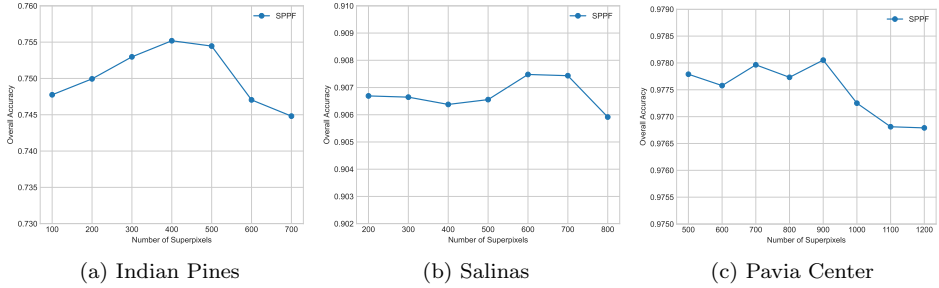
Figure 5.1: Preliminary experiment. Overall accuracies for the different datasets with 20 bands selected.

Dataset	σ	# Superpixels	w_{sp}
Indian Pines	0.5	400	1,000
Salinas	0.5	600	1,000
Pavia Center	0.5	900	10,000

Table 5.1: Hyperparameters for superpixel segmentation of the different datasets.

For the band clustering process of SPPF, a combination of grid search with trial and error was leveraged to identify suitable hyperparameters. Essential parameters for controlling the speed of convergence, such as the population size and inertia weight, were chosen with both performance and efficiency in mind, striking a good balance between the two. Particular emphasis was placed on tuning parameters w_e and w_{ct}, as those are critical parameters directly affecting output from the BS model. The hyperparameters identified during preliminary testing are presented in Table 5.2.

For comparison with the PSO-FCM model, hyperparameters were selected based on the recommendations from the original paper. The selected hyperparameters are presented in Table 5.3.

5.2   Experimental Plan

The experiments are organized into two phases. The first phase explores important design decisions of SPPF, while the second phase focuses on how the refined algorithm compares with other relevant methods. In addition, hypotheses are developed for each experiment to clarify its objectives and enhance the interpretation of the results. With the exception of Experiment 1, each experiment will

Parameter	Description	Value
pop_size	Population size	100
max_iter	Maximum iterations	100
w	Inertia weight	0.7
c_1	Cognitive component	1.0
c_2	Social component	1.0
v_{max}	Max. velocity fraction	0.1
m	Fuzzy exponent	2.0
w_e	Entropy weight	0.35
w_{ct}	Central Tendency weight	1.0

Table 5.2: Resulting hyperparameters from preliminary testing. All parameters remain the same for every run of SPPF.

Parameter	Description	Value
pop_size	Population size	100
m	Fuzzy exponent	2.0
w	Inertia weight	0.6
c	Cognitive and Social weight	1.75

Table 5.3: Hyperparameters employed by the PSO-FCM model [Zhang et al., 2017].

be evaluated on three separate datasets (Indian Pines, Salinas, and Pavia Center
- detailed in Section 5.3).

5.2.1   Phase 1: Explore design decisions of SPPF

The first phase aims to analyze how different design choices of SPPF affect clas-
sification performance and efficiency. The goal is to learn how the algorithm
can best leverage the spatial information from superpixels. The experiments
are conducted sequentially, as the implementation decisions for each subsequent
experiment depend on the results of the preceding experiment. Note that for
Experiment 1, SPPF will be utilizing SP Euclidean as the distance metric and
Highest Entropy as the criterion for cluster representative selection (as detailed
in Section 4.2.3 and Section 4.3). This aspect of SPPF will be further explored
in the following experiments.

Experiment 1 Investigate how manual pre-removal of noisy bands influences
the performance of SPPF. The experiment compares the classification ac-
curacy of SPPF using both corrected (with pre-removal of noisy bands) and
not corrected (without pre-removal) versions of two datasets (Indian Pines,
Salinas).

Hypothesis 1: The removal of noisy bands is anticipated to enhance the pre-
cision of the superpixel segmentation, resulting in improved classification.
Although the highest-entropy selection criteria will likely reduce the chance
of any noisy bands being selected, the absence of noisy bands is expected to
generally improve the band clustering process, as there are fewer destructive
outliers, resulting in potentially higher classification accuracy. The Indian
Pines Dataset is expected to show the most significant improvement from
removing noisy bands, as it is a more challenging dataset overall due to its
low spatial resolution and few labeled samples.

Experiment 2 This experiment aims to find the optimal cluster representa-
tive selection method for each distance metric applied to SPPF, based on
classification performance. Four different cluster selection methods in con-
junction with four different distance measures (as described in Section 4.3
and Section 4.2.3) will be tested with SPPF.

Hypothesis 2: The entropy-based and hybrid selection techniques (SP KL-
Divergence, SP Disjoint Information, and SP KL-Divergence-L1Norm) are
expected to perform well with distance metrics that organize a clustering
based on relative entropy, as the highest entropy bands from each cluster
should have little redundancy. In general, the weighted-sum hybrid cri-
terion is expected to perform well for all distance metrics, as its weights

have been tuned to strike a good balance between the entropy and central tendency criteria.

Experiment 3 Investigate how different distance measures can utilize the spatial information from the superpixel segmentation, and how these distance measures affect classification performance and efficiency. Four different distance measures (explained in Section 4.2.3) with resulting optimal cluster selection from Experiment 2 will be applied to SPPF to select a preferred distance metric for the SPPF model.

Hypothesis 3: The SP KL-Divergence-L1Norm distance metric is expected to achieve the best performance with the SPPF model due to its more nuanced representation of the spectral and spatial information. However, the high computational costs of the SP KL-Divergence-L1Norm distance metric are expected to make it significantly less efficient than the other proposed distance metrics.

5.2.2 Phase 2: Comparison with baseline and state-of-the-art algorithm

The second phase aims to determine whether the refined SPPF is advantageous compared to the method it is built upon and see if its performance can compete with the baseline of classifying using all available spectral bands.

Experiment 4 Determine how the performance of SPPF compares to the PSO-FCM model. Investigate whether utilizing superpixel information can improve classification performance and how it affects efficiency.

Hypothesis 4: The hypothesis suggests that the SPPF model could surpass PSO-FCM in performance by employing both spatial and spectral attributes in band clustering. The efficiency of SPPF is expected to depend on design decisions based on results from Phase 1: If employing SP-mean-based metrics, incorporating superpixel data in SPPF is expected to reduce processing time compared to PSO-FCM due to fewer necessary computations. Alternatively, if SPPF employs SP KL-Divergence-L1Norm distance, the complex overhead computations are expected to lower the computational efficiency compared to PSO-FCM.

Experiment 5 Determine how SPPF's classification performance compare to the baseline (the results obtained by using all of the original spectral bands). The aim is to identify whether SPPF can enhance classification performance while reducing the dimensionality of the HSI.

Hypothesis 5: The performance of SPPF is expected to be comparable or su-
          perior to the baseline while significantly reducing the number of required
          bands for classification.

5.3   Experimental Setup

This section introduces the datasets employed in this study, along with the evalua-
tion criteria considered. Additionally, the section introduces any non-experiment-
specific implementation details that are required for reproduction.

     The experiments conducted in this work are run on a 6-core 3.2GHz CPU
with 32GB of RAM, using the hyperparameters presented in Table 5.2, along
with any additional parameters defined within each experiment.

5.3.1   Datasets

The SPPF model's ability to perform well under different spatial, spectral, and
geographical conditions will be examined through experiments conducted on three
well-known hyperspectral datasets that have established benchmarks for classifi-
cation tasks. These datasets, which are listed in Table 5.4, were recorded by two
separate sensors in three distinct locations, and were retrieved from Grãna et al.
[2014]. Two artifacts, namely the HSI and the ground truth, are included in each
dataset. The HSI is a tensor representing the irradiance values (spectral inten-
sity) detected at the sensor's focal point. Each image pixel represents a vector of
irradiance values at a specific spatial coordinate across the available bands. The
variation in spatial resolution and the distribution of ground objects across the
three HSI datasets are key characteristics, as one of the main features of SPPF
is the utilization of spatial information.

Dataset	Dimensions	# bands	Sensor	# classes
Indian Pines	145 x 145	220	AVIRIS	16
Salinas	512 x 217	220	AVIRIS	16
Pavia Center	715 x 1096	102	ROSIS	9

Table 5.4: The datasets employed in this study.

Indian Pines

The Indian Pines dataset, captured by the AVIRIS sensor, is perhaps the most
commonly applied dataset for hyperspectral BS and classification. It covers a

scene of 145 x 145 pixels with a spatial resolution of 20 m per pixel. The scene consists of two-thirds agriculture and one-third forest or other natural perennial vegetation, and the ground truth available is designated into sixteen classes. The Indian Pines dataset has a relatively small size and low spatial resolution; thus, the land-cover regions are difficult to distinguish.

The small number of labeled samples (10,249) is a particular challenge for this dataset, as the class with the lowest amount of samples is left with only 20 samples available for both training and validation. Another interesting point about the dataset is its high variation in entropy, as can be seen in Figure 5.3 (a). Certain bands lying in the spectrum of water absorption (bands [104-108], [150-163], and 220) are deemed noisy, and are typically removed as a pre-processing step. These noisy bands are further discussed in Experiment 1 (Section 5.4.1).

Salinas

The Salinas dataset, similarly captured by the AVIRIS sensor, predominantly consists of agricultural usage. The Salinas dataset differs from Indian Pines by offering a higher spatial resolution and a bigger image size, covering an area of 512 x 217 pixels with a spatial resolution of 3.7 m per pixel. The Salinas scene also suffers from the same high variation in entropy between bands, which can be seen in Figure 5.3 (b). Similarly to Indian Pines, the Salinas dataset also has an already defined set of noisy bands that are typically removed (Bands [108-112], [154-167], and 224).

Pavia Center

The third and final dataset employed in this study is the Pavia Center dataset. The dataset initially had an image size of 1096 x 1096 pixels and a spatial resolution of 1.3 meters. However, some samples in the image lack information, as seen in Figure 5.2, where a portion in the middle of the image is black. These incomplete samples had to be removed before analysis [Grãna et al., 2014]. As a result, the image size employed for analysis was reduced to 715 x 1096 pixels, as stated in Table 5.4. This missing region of the image provides a particular challenge for BS models that rely on spatial relationships, such as SPPF, but also provides a valuable insight into how such "missing" regions affect BS models and classification.

Unlike the other datasets, the Pavia Center dataset has fewer classes and many labeled samples, and its ground truth defines nine classes. These classes include urban land use features such as asphalt, tiles, and bare soil. Additionally, the ROSIS sensor employed in this dataset distinguishes it from the AVIRIS sensors in terms of spectral dimension. Since ROSIS only covers a portion of the spectral range, information-theoretic quantities like entropy exhibit different shapes than

Figure 5.2: Sample Band Pavia Center Dataset [Grãna et al., 2014].

the AVIRIS datasets. Therefore, band information measures like KL-Divergence will demonstrate distinct properties from the other datasets, highlighting the importance of investigating this dataset.

(a) Indian Pines   (b) Salinas   (c) Pavia Center
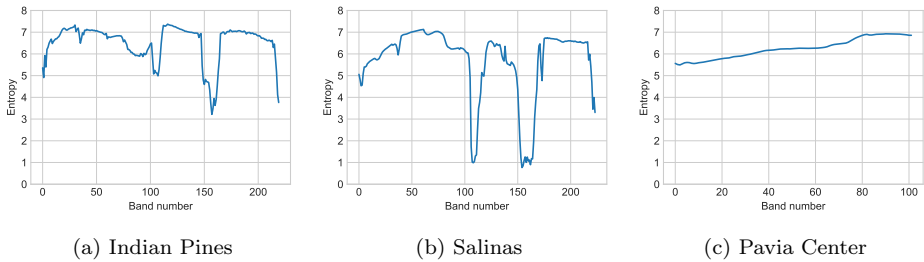
Figure 5.3: Entropy profiles for the different datasets employed.

5.3.2 Classification

After a BS model has selected bands, the resulting selected bands are then utilized to classify the dataset to evaluate the accuracy of the chosen subset. For classification, each labeled pixel in the dataset is treated as a sample, and the corresponding ground truth serves as the supervised label. In this study, only the spectral dimension is considered for classification, partly to guarantee a fair comparison to existing BS methods and partly due to spectral-spatial classification being outside the scope of this work.

Since the emphasis of this work is on the performance of the BS model rather than the classifier, the classifier itself will not be optimized for any particular BS model. This means that the achieved accuracies for a specific BS model may not be fairly compared to those achieved on a different classifier. To that end, Support Vector Machine (SVM) is selected as the classification algorithm due to its ability to perform well with small training sets, its publicly available high-quality implementation, and its reasonable computational requirements. The radial basis function will be employed as the kernel, and the kernel parameters C and γ will be established using grid search and five-fold cross-validation of the training set, as recommended by Hsu et al. [2003]. This parameter search is repeated for every band subset classified, to ensure a fair comparison.

For each band subset obtained from the BS algorithm, the classifier will typically perform multiple evaluation runs specified in the experiment specifications for each experiment. For each evaluation run, the labeled samples are randomly split into a training and testing set. Because there is a large discrepancy in available classes for the different samples, each class is split individually, guaranteeing that the same percentage of training samples is available for each class. The percentage of training samples for each dataset is specified in Table 5.5. The samples are also normalized per feature to a value between 0 and 1, a recommended pre-processing step for the SVM algorithm that ensures equal weight for each feature in the classification process. Importantly, the normalization is performed with both the training and testing set together.

Dataset	Samples	Labeled samples	Training %
Indian Pines	21,025	10,249	10%
Salinas	111,104	54,129	2%
Pavia Center	783,640	148,152	1%

Table 5.5: Training statistics for the different datasets.

5.3.3   Evaluation Metrics

When comparing BS models, it is critical to adopt robust and reliable metrics to measure their performance. This subsection presents and discusses the metrics employed for evaluation in our analysis.

Classification accuracy

The performance of the classification algorithm will be evaluated using two accuracy-based metrics. The first is overall accuracy (OA), simply the number of correctly

labeled test samples divided by the total number of test samples. Both the mean and standard deviation of this value will be recorded. The second metric is average accuracy (AA), the average fraction of correctly labeled test samples of each class. This metric can provide insight into how the selected bands perform in classifying different classes, as the OA can be easily skewed by a single class with significantly more labeled samples than the other classes.

McNemar's test

The comparison and evaluation of different BS models' classification performance necessitate the employment of a robust statistical test. In this study, McNemar's test, a non-parametric method for testing the independence of two variables, is employed to verify the statistical significance of the results observed. The test is based on the standardized normal test statistic and is particularly suited when comparing BS models [Foody, 2004].

For evaluating the significance of difference in classification between two BS models, it defines the term f_{12}, which is the number of samples correctly classified by model 1 and misclassified by model 2. Conversely, the term f_{21} is defined as the number of samples correctly classified by model 2 and misclassified by model 1. Following this, the McNemar's test statistic, which measures the difference in performance between the two models, is calculated as follows:

$$Z = \frac{f_{12} - f_{21}}{\sqrt{f_{12} + f_{21}}}, \tag{5.1}$$

Notably, a $|Z|$ value of 1.96 or higher denotes a statistically significant difference at a confidence level of 95% or higher ($p \leq 0.05$). This measure ensures that any differences in performance between the models are statistically significant.

5.4   Experimental Results

This section presents a analysis of the experimental results obtained in this study. Each experiment presents the questions it aims to answer, along with the details needed to replicate it. Results from each experiment are discussed with a particular attention to identifying trends and anomalies. Each experiment also completes with a conclusion describing the most important findings observed. Additional information for some experiment runs can also be found in Appendix A1.

5.4.1 Experiment 1 - Noisy Band Influence

This experiment investigates two critical aspects of the SPPF model: the effect of noisy bands on the BS process, and the impact of manual pre-removal of noisy bands on the classification accuracy of the selected band subset. To explore these aspects, the SPPF model was applied to two datasets, Indian Pines and Salinas, using the bands left after the manual pre-removal of noisy bands, and all available bands. These datasets were selected because they have an already identified set of noisy bands, specifically those covering the water absorption region. The noisy bands for Indian Pines are bands [104-108], [150-163], and 220. For Salinas, the noisy bands are [108-112], [154-167], and 224.

The experiment was conducted by running the algorithm 180 times with all bands and 180 times without the noisy bands. These 180 runs were spread across 18 different n values (number of bands to select), with ten runs for each value. The complete experiment specifications can be seen in Table 5.6. The resulting performance of the BS algorithm is shown in Figure 5.4, Figure 5.7, and Table 5.7.

Experiment Specifications	
Model	SPPF
Dataset	Indian Pines, Indian Pines (noisy bands removed), Salinas, Salinas (noisy bands removed)
Number of bands selected (n)	2, 3, 4, 5, 6, 7, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30
Runs	10
Stopping criterion	100 generations OR 15 generations without improvement
Number of classification runs	10
Distance metric	Superpixel Euclidean distance
Cluster selection	Highest entropy

Table 5.6: Experiment 1 - Specifications.

For the Indian Pines dataset, the results shown in Figure 5.4 indicate that the version without noisy bands demonstrates superior overall performance with respect to both Overall Accuracy and Average Accuracy, especially when the number of selected bands increases. When selecting fewer bands, specifically less than 15, both versions exhibited some instability, and no version convincingly outperformed the other in these cases. This behavior is expected, as the inclusion or omission of a singular high-entropy band to a cluster has a more significant impact when the number of chosen bands is low.

(a) Overall accuracy   (b) Average accuracy
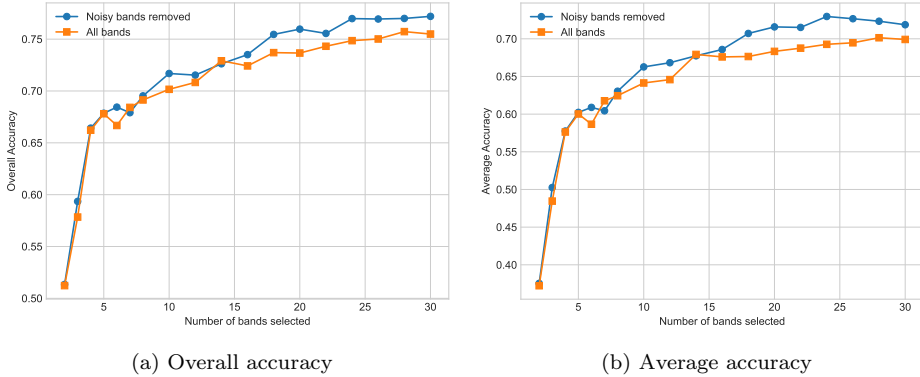
Figure 5.4: Experiment 1. Performance results on the Indian Pines dataset.

Comparing the rows for Indian Pines in Table 5.7, it is noteworthy that the mean overall standard deviation (SD) is slightly worse (higher) for the version with the pre-removal of noisy bands. This can be attributed to the most considerable differences in SD occurring when a low number of bands are selected, specifically when three bands are selected, as seen in Figure 5.5. In these cases, the version with manual band removal is able to occasionally select a significantly better subset of bands, while the version using all bands is consistently poor. This is shown in Figure 5.6, where the two runs at the top achieve significantly better overall accuracy, as shown by the numerical values on the left. However, as seen in Table 5.7, when considering runs where at least 15 bands are selected, the version using manual pre-removal of noisy bands has a slightly better (lower) SD than the version using all bands. This implies that the dataset with noisy bands yields more consistent results than the version employing all bands.

| Dataset | All runs | | | | Runs with $n \geq 15$ | | | |
	OA	OA SD	AA	AA SD	OA	OA SD	AA	AA SD
Indian Pines	0.698	**0.013**	0.630	**0.020**	0.744	0.015	0.689	0.021
Indian Pines (NBR)	**0.708**	0.014	**0.646**	0.021	**0.761**	**0.014**	**0.715**	**0.019**
Salinas	**0.879**	0.006	**0.905**	0.009	0.907	0.003	0.943	0.003
Salinas (NBR)	0.877	**0.005**	0.899	**0.007**	**0.908**	**0.002**	**0.944**	**0.002**

Table 5.7: Experiment 1 - Aggregated results (mean) for Indian Pines and Salinas datasets. Datasets with a suffix of (NBR) have had the noisy bands removed prior to the run.
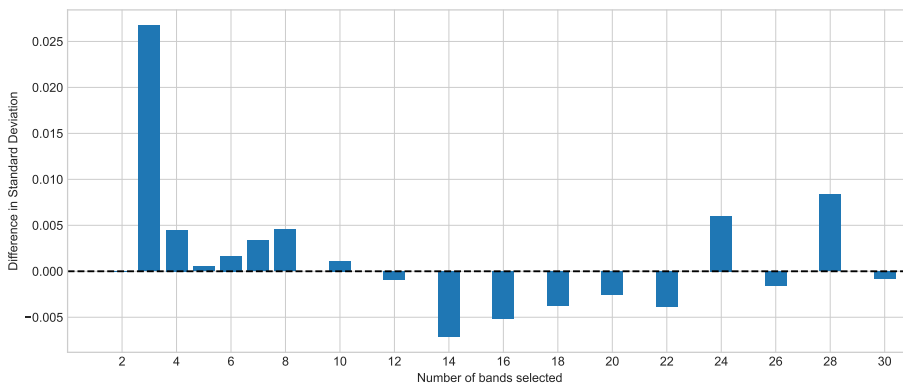
Figure 5.5: Experiment 1. Difference in SD of Overall Accuracy on the Indian Pines dataset. Positive values indicate that the version with pre-removal of bands has a higher SD than the version using all bands.
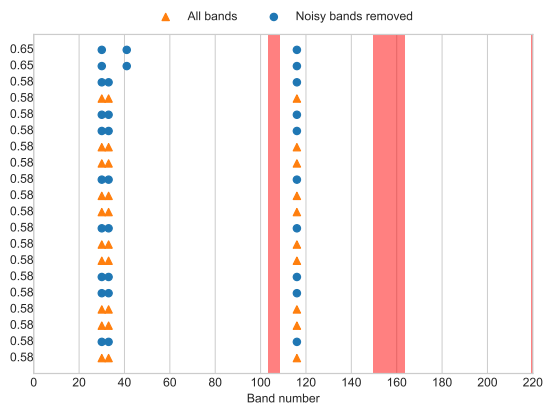
Figure 5.6: Experiment 1. Distribution of selected bands for all runs of Indian Pines when $n = 3$. Each row is a separate run. The red rectangles represent the noisy bands, which are only available for selection for the model without pre-removal of noisy bands. The metric on the far left is the achieved Overall Accuracy for those selected bands.

As for the results on the Salinas dataset, displayed in Figure 5.7, the dataset containing all bands outperformed the version with noisy bands removed when the number of selected bands was low ($n \leq 5$). For higher n-values, both datasets perform quite similarly with no significant differences in achieved performance. Table 5.7 shows that the version of the dataset with pre-removal of noisy bands shows better consistency overall, with always having a lower SD.

(a) Overall accuracy   (b) Average accuracy
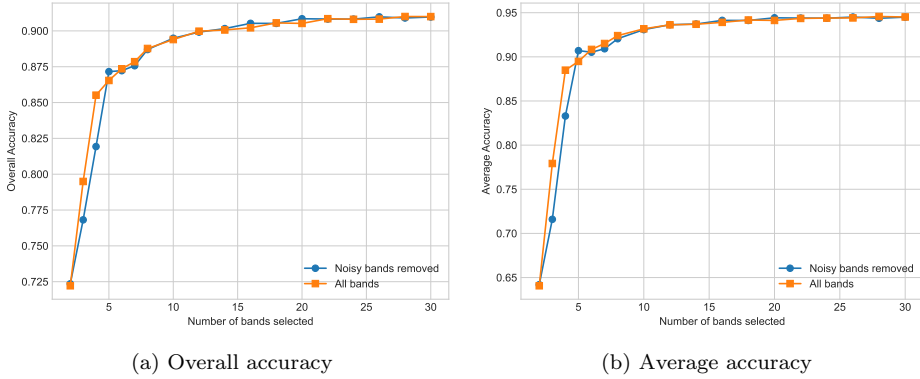
Figure 5.7: Experiment 1. Performance results on the Salinas dataset.

Conclusion

The results of Experiment 1 suggest that the SPPF model may be somewhat negatively affected by the presence of noisy bands in the datasets. Therefore, it is recommended to perform manual pre-removal of noisy bands to achieve better classification accuracy. This is especially true for higher n-values.

Based on the findings, future experiments involving the SPPF model will be performed using the versions of datasets without the noisy bands. This approach is expected to yield better and more consistent results overall.

5.4.2 Experiment 2 - Cluster Representative Selection

The second experiment investigates different ways of choosing cluster representatives to identify the optimal cluster selection method for each distance metric applied to the SPPF model. The experiment evaluates the Overall Accuracy and Average Accuracy of four distance measures, namely SP Euclidean distance, SP KL-Divergence, SP Disjoint Information, and SP KL Divergence-L1Norm, on three hyperspectral datasets: Indian Pines, Salinas, and Pavia Center. Four cluster representative selection techniques are explored to find the optimal cluster selection method: Highest Entropy, Central Tendency, Hybrid Weighted-Sum Criterion, and Hybrid Ranking Criterion. Each cluster selection method is applied to each distance measure, and the resulting accuracies for the different datasets are evaluated. The complete experiment specifications can be seen in Table 5.8.

Experiment Specifications	
Model	SPPF
Dataset	Indian Pines, Salinas, Pavia Center
Number of bands selected (n)	5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Runs	10
Stopping criterion	100 generations OR 15 generations without improvement
Number of classification runs	10
Distance metric	SP Euclidean distance, SP KL-Divergence,
	SP Disjoint Information, SP KL Divergence-L1Norm
Cluster selection	Highest entropy, Central tendency,
	Hybrid weighted-sum criterion, Hybrid ranking criterion

Table 5.8: Experiment 2 - Specifications.

SP Euclidean Distance

For the SP Euclidean Distance metric, experimental findings displayed in Table 5.9 show a close competition between the Central Tendency and Weighted-sum hybrid selection methods. The Central Tendency method displays slightly better performance on the Indian Pines and Salinas datasets, while the Weighted-sum hybrid method outperforms it marginally on the Pavia Center dataset. Due to its superior overall performance, the Central Tendency criterion is chosen as the preferred selection method for the SP Euclidean Distance metric.

Interestingly, all selection criterions demonstrated similar performance levels on the Pavia Center dataset, with the Highest Entropy method emerging as the winner by a slim margin. This outcome is surprising, considering the Highest

Entropy method performs significantly worse than all other selection methods for the Indian Pines and Salinas dataset.

Among the hybrid selection methods, the Ranking hybrid consistently performed worse than the Weighted-sum hybrid criterion. This observation suggests that the Weighted-sum hybrid is a more reliable option when using hybrid methods in combination with the SP Euclidean Distance metric.

	Indian Pines		Salinas		Pavia Center	
	OA	AA	OA	AA	OA	AA
Highest Entropy	0.758	0.706	0.903	0.939	**0.976**	**0.923**
Central Tendency	**0.779**	**0.736**	**0.909**	**0.944**	**0.976**	0.921
Weighted-sum	0.778	0.734	0.908	**0.944**	**0.976**	0.922
Ranking	0.771	0.720	0.907	0.943	0.975	0.921

Table 5.9: Experiment 2. Aggregated results for all datasets using the SP Euclidean Distance metric. The highlighted results are the best for that column.

SP KL-Divergence Distance

Table 5.10 presents the aggregated results for all datasets using the SP KL Divergence distance metric. The Weighted-sum hybrid selection method is found to be the overall winner, performing on par or better than all other selection methods for all datasets. The only exception was a slight disadvantage regarding average accuracy for the Pavia Center dataset. This suggests that the Weighted-sum hybrid is an effective and consistent criterion for selecting cluster representatives when using the SP KL Divergence distance metric.

As shown in Table 5.10, the Central Tendency criterion performed on par with the Weighted-sum hybrid criterion for the Salinas and Pavia Center datasets. However, it was significantly inferior to the Weighted-sum hybrid criterion for the Indian Pines dataset. Interestingly, the difference between the best and second-best criterion for the Indian Pines dataset under the SP KL Divergence distance metric was larger than any other distance metric. The difference was at least 3x larger for OA and 1.5x larger for AA compared to other distance metrics. This observation highlights the unique advantage of combining the SP KL Divergence distance metric and the Weighted-sum hybrid criterion.

Based on the results presented in Table 5.10, the Weighted-sum hybrid criterion was selected as the best option for the SP KL Divergence distance metric in SPPF. This combination yields strong performance across various datasets, with a particularly notable advantage for the Indian Pines dataset.

	Indian Pines		Salinas		Pavia Center	
	OA	AA	OA	AA	OA	AA
Highest Entropy	0.745	0.689	0.899	0.936	0.974	0.918
Central Tendency	0.758	0.710	**0.902**	**0.939**	**0.975**	0.920
Weighted-sum	**0.765**	**0.716**	**0.902**	**0.939**	**0.975**	0.920
Ranking	0.754	0.697	0.897	0.934	**0.975**	**0.921**

Table 5.10: Experiment 2. Aggregated results for all datasets using the SP KL-Divergence Distance metric. The highlighted results are the best for that column.

SP Disjoint Information

For the SP Disjoint Information distance metric, the experimental findings shown in Table 5.11 revealed a close competition between the Central Tendency and Weighted-sum hybrid selection methods. Both methods demonstrate comparable performance, with none of the two being the clear preferred choice. In contrast, the Ranking and Highest Entropy selection methods are found to be significantly worse overall. However, it is interesting to note that the Ranking criterion exhibited the best performance on the Pavia Center dataset, tied with the Central Tendency criterion. This observation suggests that the Ranking criterion may have specific advantages under certain conditions.

Due to its marginally superior overall performance, the Central Tendency selection method was chosen as the preferred option for the SP Disjoint Information Distance Metric.

	Indian Pines		Salinas		Pavia Center	
	OA	AA	OA	AA	OA	AA
Highest Entropy	0.734	0.678	0.896	0.930	0.970	0.905
Central Tendency	**0.742**	**0.679**	0.899	0.936	**0.975**	**0.919**
Weighted-sum	0.740	**0.679**	**0.900**	**0.937**	**0.975**	0.918
Ranking	0.733	0.665	0.893	0.929	**0.975**	**0.919**

Table 5.11: Experiment 2. Aggregated results for all datasets using the SP Disjoint Information Distance metric. The highlighted results are the best for that column.

SP KL Divergence-L1Norm

For the KL Divergence L1-Norm distance metric, experimental findings displayed in Table 5.12 show that the Central Tendency method is the best overall choice for selecting cluster representatives. This method demonstrates superior performance for the Indian Pines and Salinas datasets, while being narrowly beaten by the Weighted-sum method for the Pavia Center dataset. The Weighted Sum and Ranking methods exhibit relatively even performance, and although they are not as good as the Central Tendency method, they still deliver competitive results, particularly for the Pavia Center dataset. In contrast, the Highest Entropy method is shown to be the least effective option for the KL Divergence L1-Norm distance metric. It consistently records the lowest overall and average accuracies across all three datasets.

Based on these findings, the Central Tendency method has been selected as the best choice for the KL Divergence L1-Norm distance metric in SPPF.

	Indian Pines		Salinas		Pavia Center	
	OA	AA	OA	AA	OA	AA
Highest Entropy	0.767	0.717	0.901	0.936	0.975	0.921
Central Tendency	**0.782**	**0.741**	**0.907**	**0.942**	**0.976**	0.923
Weighted-sum	0.779	0.735	0.905	0.941	**0.976**	**0.924**
Ranking	0.780	0.737	0.903	0.938	**0.976**	0.923

Table 5.12: Experiment 2. Aggregated results for all datasets using the SP KL Divergence-L1Norm Distance metric. The highlighted results are the best for that column.

Conclusion

The analysis of experimental findings for the SPPF BS model using various distance metrics and representative selection methods has provided valuable insights into the effectiveness of different combinations. Generally, the Weighted-sum hybrid and Central Tendency methods both exhibit good and consistent performance across all distance metrics, suggesting that the resemblance to other bands in the cluster might be more important than always selecting the highest entropy bands.

A summary of the final preferred selection methods for each distance metric is presented in Table 5.13. These selection methods have been chosen based on their superior performance in the respective distance metric context. Future

experiments involving these distance metrics with the SPPF model will employ the given selection method.

Distance Metric	Selection Method
SP Euclidean Distance	Central Tendency
SP KL-Divergence Distance	Weighted-sum Hybrid
SP Disjoint Information	Central Tendency
SP KL Divergence-L1Norm	Central Tendency

Table 5.13: Experiment 2. Preferred cluster representative selection method for each distance metric.

5.4.3   Experiment 3 - Distance Measures

The following experiment investigates how the different distance measures utilize the superpixel map and their impact on the classification accuracy and efficiency when applied to SPPF. Similarly to Experiment 2, the distance metrics applied include SP Euclidean distance, SP KL-Divergence, SP Disjoint Information, and SP KL Divergence-L1Norm distance (all described in Section 4.2.3). These distance metrics will employ their evidently preferred selection methods, as presented in Table 5.13. The performance of the different metrics is evaluated on the Indian Pines, Salinas, and Pavia Center datasets. Furthermore, different numbers of bands, n, ranging from 5 to 50, are selected for analysis. The complete experiment specifications can be seen in Table 5.14.

Experiment Specifications	
Model	SPPF
Dataset	Indian Pines, Salinas, Pavia Center
Number of bands selected (n)	5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Runs	10
Stopping criterion	100 generations OR 15 generations without improvement
Number of classification runs	10
Distance metric	SP Euclidean distance, SP KL-Divergence, SP Disjoint Information, SP KL Divergence-L1Norm

Table 5.14: Experiment 3 - Specifications.

Comparison of Classification Accuracy - Indian Pines

Figure 5.8 presents the results of the overall accuracy (OA) and average overall accuracy (AA) for all distance measures applied to the Indian Pines dataset. The results show that the information-theoretic distance measures (SP KL-Divergence, SP Disjoint Information, and SP KL-Divergence-L1Norm) exhibit considerably lower accuracy than SP Euclidean distance when few bands are selected ($n = 5$), with a rapid improvement in the range between five and fifteen bands selected.

This may be because the variability of the entropy can be significant, resulting in distance metrics that rely on relative entropy exhibiting considerable variability across different bands. This variability may become problematic when forming clusters based on information-theoretic measures, especially when the number of clusters is small. Outlier bands, with high differences in entropy, can disproportionately influence the selection of representative bands for each cluster. These outlier bands may significantly distort the result in determining cluster representatives through criteria such as Central Tendency or Weighted-sum.

Consequently, the bands chosen as representatives may not accurately reflect the majority of bands within each cluster; instead, they tend to be those with high entropy. In essence, these chosen representatives can end up being outlier bands within their clusters due to their high entropy.

As a result, the final selection may favor redundant bands if the number of clusters is insufficient, as demonstrated in Figure 5.9 (a). Figure 5.9 illustrates the bands selected (green) and their respective entropy values (blue) in the SPPF approach utilizing SP KL Divergence L1Norm on Indian Pines. In Figure 5.9 (a), when ten bands are selected ($n = 10$), the selected bands are primarily those with high entropy and are almost adjacent to each other. In contrast, when fifteen bands ($n = 15$) are selected, the bands are more evenly distributed across all available bands, primarily avoiding dips in entropy, ss shown in Figure 5.9 (b). This suggests that the impact of outlier bands is diluted across a larger number of clusters, thereby mitigating its influence. Thus, the increase in classification accuracy from 10 to 15 bands for the information-based measures could be attributed to this dilution of outlier influence.

Figure 5.8 also reveals that the Superpixel level KL-L1Norm distance metric demonstrates superiority at $n > 15$, while the SP KL-Divergence exhibits more instability. These findings indicate that considering the relative entropy of each superpixel between bands is advantageous, compared to only utilizing the relative entropy of the aggregated mean values, such as SP KL-Divergence. Another interesting observation for the SP KL-Divergence metric is that at $n = 30$, both OA and AA display a noticeable decline, followed by a subsequent increase at $n = 35$. The same pattern can be observed for SP Disjoint, which is expected since Disjoint information is related to the KL-Divergence (detailed in Section 2.2) and both measure the divergence between two bands' probability distributions based on the mean values of the superpixels, rather than individual pixels. The results on the Indian Pines dataset also show that the SP Disjoint Information metric was significantly outperformed by all other distance metrics. This indicates that the probability distributions based on the mean values of the superpixels were too divergent, affecting SP Disjoint Information to a greater extent than SP KL-Divergence. This is because the joint probability used to calculate SP Disjoint Information is more severely affected. More specifically, when the superpixel's mean values are too divergent, it means the likelihood of these values occurring together becomes less likely. This results in a greater impact on the SP Disjoint Information, which relies heavily on these joint probabilities in contrast to SP KL-Divergence which only measures how one probability distribution diverges from another.

(a) Overall accuracy   (b) Average accuracy
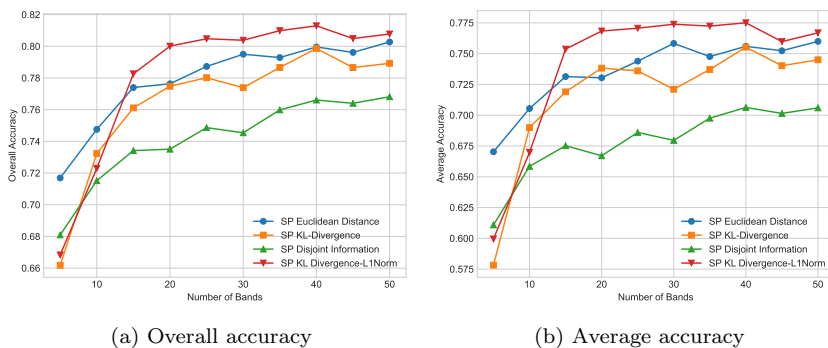
Figure 5.8: Experiment 3. Classification performance Indian Pines.

(a) Ten bands   (b) Fifteen bands
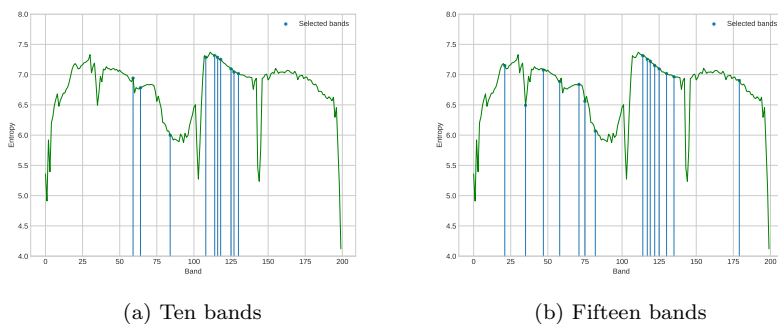
Figure 5.9: Selected bands of SPPF employing SP KL-Divergence-L1Norm on the Indian Pines dataset.

(a) Overall accuracy   (b) Average accuracy
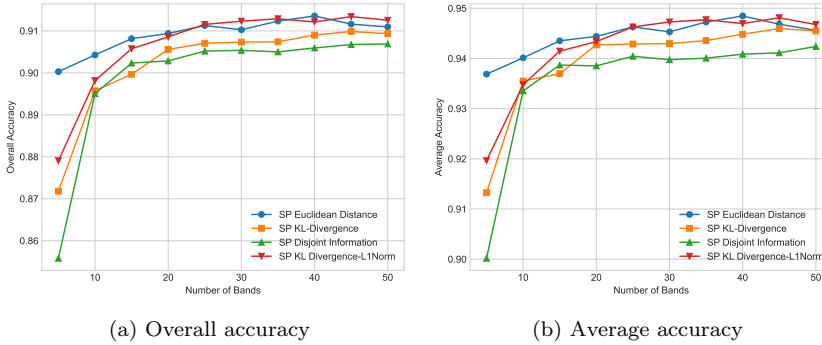
Figure 5.10: Experiment 3. Classification performance Salinas.

Comparison of Classification Accuracy - Salinas

Figure 5.10 displays the results obtained for the Salinas dataset, which shows comparable patterns when bands selected are less than fifteen ($n < 15$), as observed for the Indian Pines dataset. Again the performance of measures relying on information quantities is poor for low n-values. This supports the above explanation (Subsection 5.4.3).

In contrast to the results of the Indian Pines dataset, Figure 5.10 shows the SP Euclidean Distance metric is able to compete with the SP KL-Divergence-L1Norm metric for higher n-values, in terms of both OA and AA. The two display a superior performance to the other distance metrics, with SP Disjoint Information again displaying the worst performance overall. It is worth noting that the overall differences are much more minor than on the Indian Pines dataset, with only about a single percentage point separating the best and the worst metric for higher n-values. These observations also suggest that the SP-mean values are more representative of every pixel within a superpixel in the Salinas dataset, making the calculation of divergence between each superpixel less advantageous than the other distances utilizing SP-mean values. Given that the Salinas dataset has better spatial resolution than Indian Pines, it can be expected that the superpixel segmentation could be more accurate.

Comparison of Classification Accuracy - Pavia

The findings for the Pavia Center dataset, as presented in Figure 5.11, reveal that all distance metrics employing mean values (SP Euclidean, SP KL-Divergence, SP Disjoint) show only marginal differences in OA-values and AA as the number of bands selected increases beyond 20. This observation can be attributed to

the dataset being relatively easy to classify, owing to the smaller number of
classes combined with high spatial resolution. Interestingly, the application of the
SP-KL-Divergence-L1norm leads to a markedly different behavior, with SPPF
exhibiting a significant peak in classification accuracies at 30 bands ($n = 30$).
Upon examining the selected bands for different measures, it can be observed
from Figure 5.12 that SPPF employing SP KL-Divergence L1norm demonstrates
a tendency to select fewer of the lowest entropy bands, as it favors the selection
of high entropy bands. As shown in Figure 5.12 (a), SP-KL-Divergence-L1norm
(red) consistently has higher average entropy across selected bands compared to
SP Euclidean (blue). It is important to note that when the number of selected
bands increases, the average entropy decreases because it becomes more likely
to select bands with lower entropy. When looking closer into which bands are
selected for $n = 30$, shown in Figure 5.12 (b), it can be seen that SP Euclidean
is characterized by a more even dispersion of selected bands, so more bands of
lower entropy are selected, as confirmed in (a). The higher entropy combined
with the sufficient spread between the selected centroids explains why SP KL-
Divergence L1norm may have achieved superior classification accuracy at this
point. The SP-KL-Divergence-L1norm distance metric displays a descending
trend in accuracy from 30 to 50 selected bands for the Pavia Center dataset, which
contrasts with the behavior observed for the Indian Pines and Salinas datasets
using the same distance metric. This distinct pattern in the Pavia Center dataset
may be attributed to its significantly fewer bands overall, where the selection of 50
bands at the Pavia Center represents roughly half of all available bands, whereas
50 bands for Indian Pines and Salinas is only about 25%.

(a) Overall accuracy   (b) Average accuracy
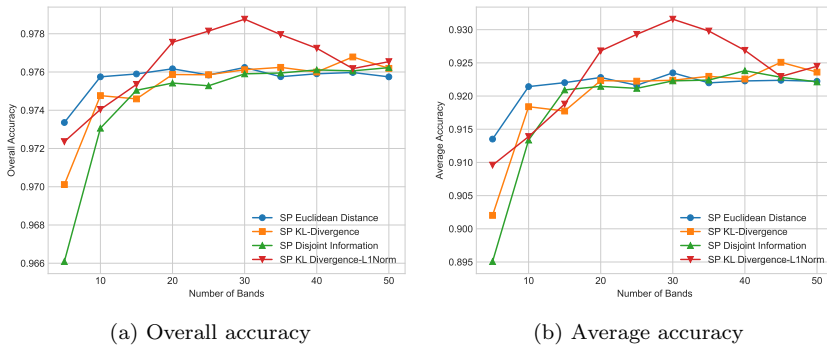
Figure 5.11: Experiment 3. Classification performance Pavia Center.
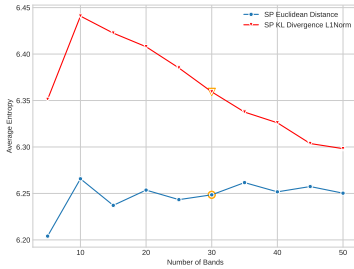
(a) Average entropy selected bands
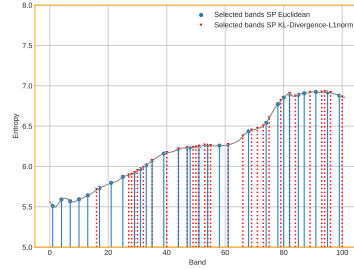
(b) Entropy 30 bands

Figure 5.12: Comparison of entropy. The point of 30 bands is highlighted in yellow (a), since (b) provides a closer look at the specific bands selected at this point.

Runtime Comparison

Figure 5.13 shows the mean runtime for a single iteration when selecting 20 bands across all distance metrics and datasets. Interestingly, the runtime for the Pavia Center dataset (in green) is significantly lower than the other datasets, even though it has a much higher spatial resolution and a higher number of superpixels. This indicates that, in general, the runtime of the SPPF model depends most heavily on the number of spectral bands available in the HSI, which is lower in the Pavia Center dataset than in the other. Figure 5.13 also shows that the SP Euclidean Distance Measure is able to compete with the runtime of the SP Disjoint and SP KL-Divergence L1-Norm measures, even though they both are pre-computed ahead of time, and the SP Euclidean measure is not.

In addition, the pre-computation of SP KL-Divergence and SP Disjoint and SP KL-Divergence L1-Norm measures causes a larger overhead, as shown in Figure 5.14, where the mean runtime for overhead calculations of each distance metric is displayed with 20 bands selected. One particularly noteworthy aspect is the significantly larger runtime for overhead calculations of SP KL-Divergence L1-Norm. However, this was expected, as a 3D distance tensor is precomputed, giving it a significantly larger overhead computational cost (as described in Section 4.2.3).

Conclusion

In conclusion, the experiment comparing distance measures showed that SP Euclidean distance is more computationally efficient than the other distance metrics
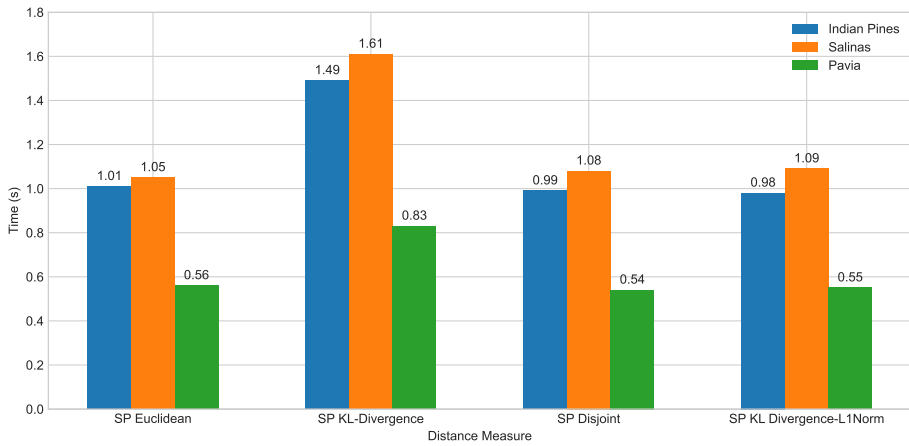
Figure 5.13: Mean runtime for a single iteration with 20 bands selected.
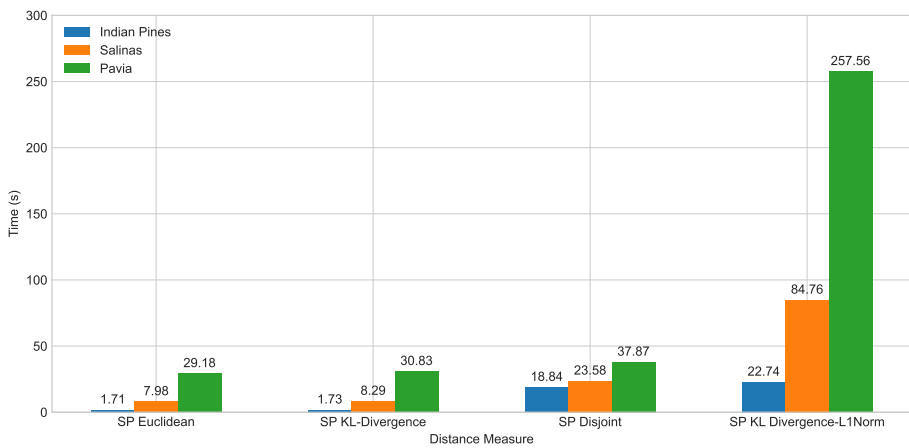
Figure 5.14: Mean runtime for overhead calculations with 20 bands selected.

(SP Disjoint, SP KL-Divergence, and SP KL-Divergence L1Norm). However, the SP KL-Divergence L1Norm distance metric employed by the SPPF algorithm demonstrated higher accuracy on both Indian Pines and Pavia Center datasets and achieved comparable accuracies on the Salinas dataset. This is particularly evident when selecting more than 15 bands, which aligns with the prioritized objective.

The results suggest there is an advantage to considering the relative entropy within each superpixel for each band, rather than utilizing the mean value. This advantage becomes more pronounced when dealing with datasets that exhibit lower spatial resolution and pose challenges in classification, as observed in the case of Indian Pines. Considering the differences within each superpixel allows for a more nuanced representation of the spectral and spatial information, especially when subtle variations within superpixels may affect the classification outcomes.

In terms of runtime, SP Euclidean appears to provide better efficiency than other measures. Interestingly, the Pavia Center dataset had lower runtime despite having a higher resolution and more superpixels, suggesting that spectral bands impact the SPPF model's runtime more than spatial resolution or superpixel count. It has also been observed that the SP Euclidean offers competitive runtime performance when compared to pre-computed measures. In contrast, the SP KL-Divergence L1-Norm distance results in significantly higher overheads and runtime as expected.

Based on these findings, future experiments with SPPF will incorporate the SP KL-Divergence L1Norm distance measure for BS. This decision is based on its superior performance across the datasets when a larger number of bands are selected.

5.4.4   Experiment 4 - Comparison with PSO-FCM

Experiment 4 examines how the SPPF model compares against the PSO-FCM model, from which it drew inspiration. Specifically, the objective is to investigate whether incorporating superpixel information enhances classification accuracy. Additionally, the two algorithms' efficiency will be compared. The complete experiment specifications are outlined in Table 5.15.

Experiment Specifications	
Model	SPPF, PSO-FCM
Dataset	Indian Pines, Salinas, Pavia Center
Number of bands selected (n)	5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Runs	20
Stopping criterion	100 generations OR 15 generations without improvement
Number of classification runs	10

Table 5.15: Experiment 4 - Specifications.

Comparison Indian Pines and Salinas

(a) Overall accuracy   (b) Average accuracy
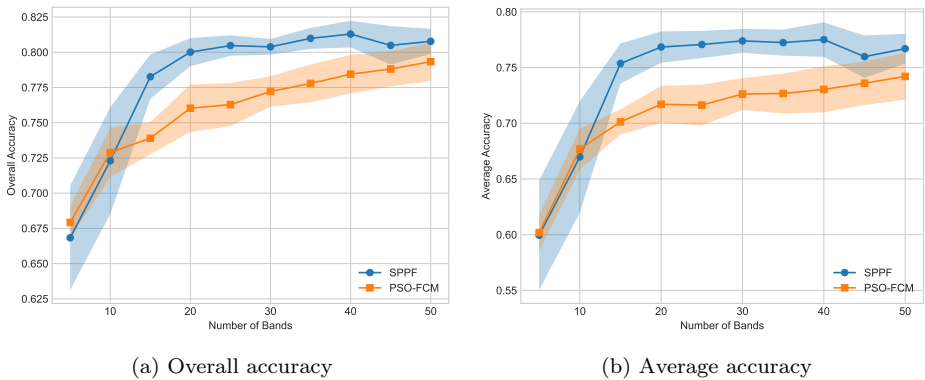
Figure 5.15: Experiment 4. Classification performance, Indian Pines dataset.

Analyzing the results on the Indian Pines datasets from Figure 5.15, it becomes apparent that the SPPF model exhibits both strengths and limitations. The SPPF model outperforms PSO-FCM significantly when the number of selected bands increases ($n \geq 15$), both in terms of higher accuracy and a slightly

lower standard deviation (as shown by the shaded region). In contrast, when $n < 15$, the SPPF's performance is slightly worse and demonstrates a significantly higher standard deviation. It is clear that the SPPF model struggles to be consistent when the number of bands selected is low. This observation aligns with the anticipated outcome derived from the findings of Experiment 3.


(a) Overall accuracy   (b) Average accuracy
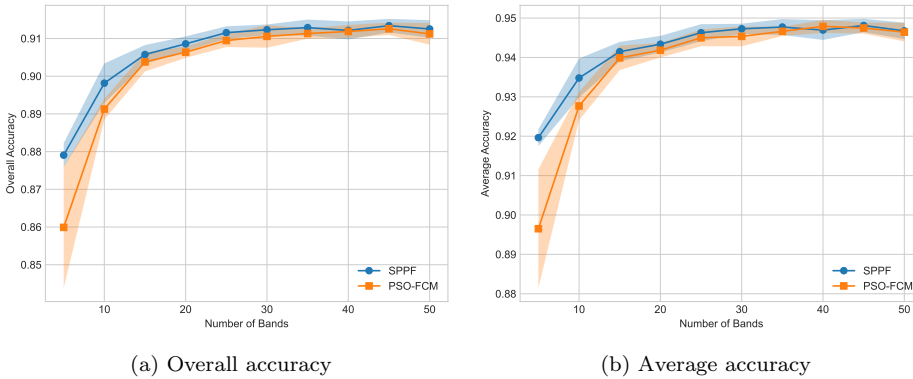
Figure 5.16: Experiment 4. Classification performance, Salinas dataset.

It is worth noting that the Salinas dataset presents a distinct scenario, as the SPPF model outperforms PSO-FCM for all selected band numbers. Figure 5.16 clearly shows a consistently lower standard deviation from the start, which can be explained by the dataset's finer spatial resolution and simpler classification attributes. As a result, integrating superpixel data can offer considerable benefits, even for a limited number of bands, without experiencing the instability seen in SPPF runs with fewer bands.

Pavia Center comparison

Interestingly, as seen in Figure 5.17, SPPF is generally outperformed by PSO-FCM for the Pavia Center dataset, except for the ranges of 20-40 bands selected. SPPF's poor performance might be due to the missing regions of the image, as described in Section 5.3.1. Such missing image data will naturally impact BS models utilizing spatial information more since it reduces the superpixel segmentation's ability to separate distinct objects into separate segments. Figure 5.18 highlights this issue by showing the superpixel segmentation on this dataset. It shows, highlighted in green, how the segmentation algorithm sometimes cannot distinguish between objects from the left and right sides of the missing image parts. The white vertical line shows the border between the left and right sides

of the image, where the missing areas of the image would lie between. The image highlights just one of many such problematic areas in the superpixel segmentation for the Pavia Center dataset. This problematic segmentation might be the reason for SPPF's poor performance for this dataset and suggests that incorporating spatial information into the BS process is less effective when the image is missing larger spatial regions of the image.


(a) Overall accuracy   (b) Average accuracy
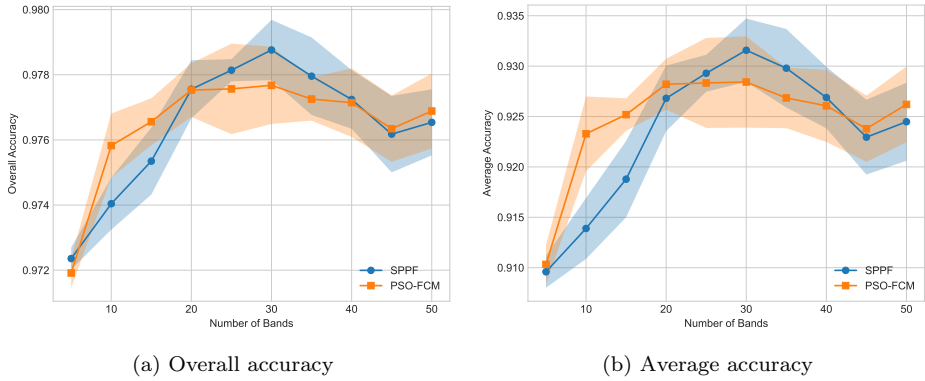
Figure 5.17: Experiment 4. Classification performance, Pavia Center dataset.
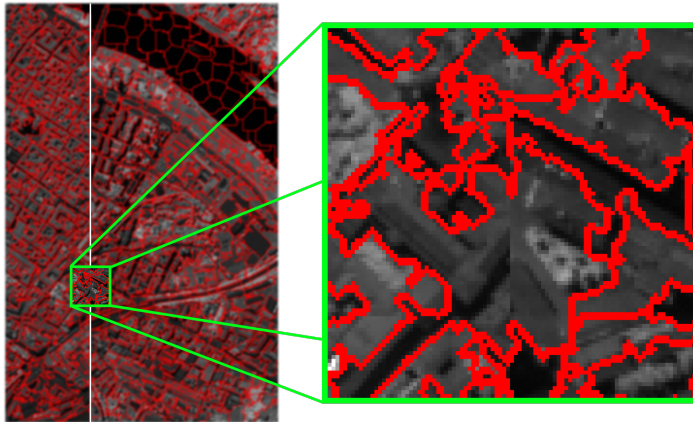
Figure 5.18: Experiment 4. Pavia Center Segmentation.

As mentioned in Section 5.4.3, the classification accuracy of SPPF can be negatively affected when only a few bands are selected, because of the low dispersion of information within the selected bands. This differs from the PSO-FCM algorithm, which employs the Euclidean distance metric. Although PSO-FCM utilizes distances between individual pixels rather than superpixel mean values, similar trends with higher degrees of dispersion can be seen in Figure 5.19. The figure displays the selected bands and their entropy at the point where the PSO-FCM algorithm demonstrates its highest superiority in terms of classification accuracy ($n = 10$).


(a) SPPF (b) PSO-FCM
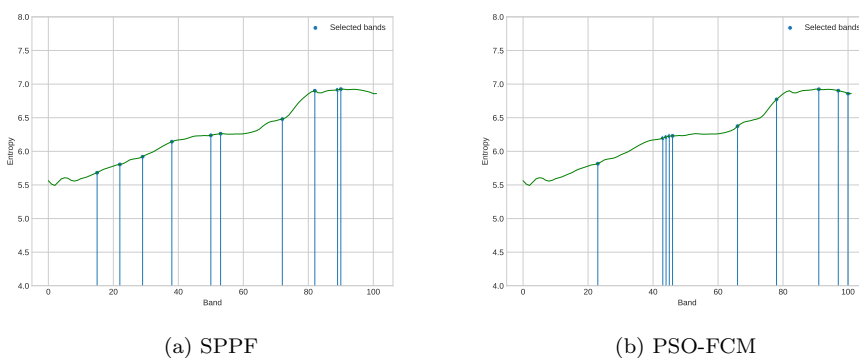
Figure 5.19: Experiment 4. Comparison selected bands ($n = 10$).

Runtime Comparison

Figure 5.20 compares the mean runtime between the SPPF and PSO-FCM models when selecting 20 bands across all datasets. The runtime of PSO-FCM, represented by the orange bars, exhibits relatively low variations across the datasets compared to SPPF, represented by the blue bars. The results show that SPPF displays inferior runtime and efficiency to PSO-FCM, with the difference more significant for the datasets with a higher number of pixels. Considering the decision to prioritize classification accuracy over efficiency when selecting the distance measure of SPPF, it was expected that the runtime of SPPF would not be favorable to PSO-FCM due to the significant overhead computations.
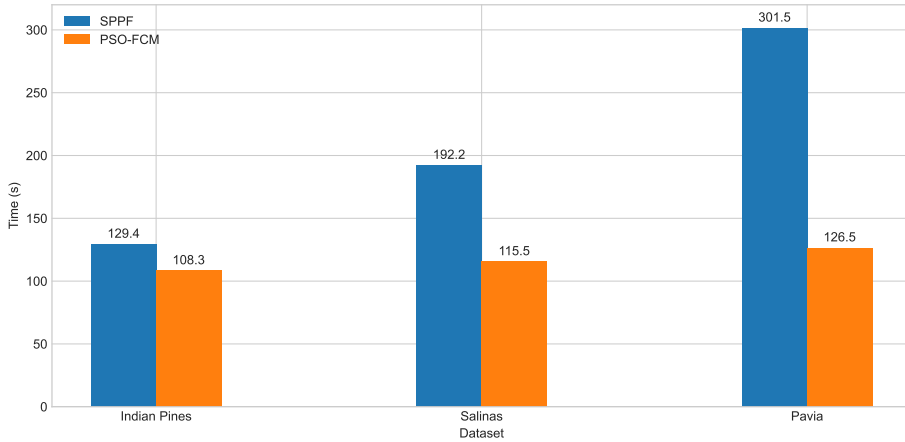
Figure 5.20: Mean total runtime for 100 iterations 20 bands selected.

Statistical Significance

To compare the performance of the SPPF and PSO-FCM models, McNemar's test is employed to assess statistical significance, as described in Section 5.3.3. The resulting Z-values from ten different trials for each combination of dataset and bands selected are presented in Table 5.16. The values are labeled Z_0 through Z_9, each representing the statistical significance of the models for one trial with the combination of dataset and n-value specified in columns 1 and 2. The corresponding selected bands for each trial can be found in Appendix A1.

Interpreting the Z-values, it is important to note that values greater than 1.96 indicate that the SPPF model significantly outperforms PSO-FCM with over 95% confidence ($p \leq 0.05$). Conversely, values lower than -1.96 suggest that PSO-FCM outperforms SPPF. Z-values falling between these thresholds indicate that no definitive conclusion can be drawn with over 95% confidence regarding one model's superiority over the other.

Table 5.16 reveals that the Indian Pines dataset exhibits a consistent trend where the SPPF model significantly outperforms PSO-FCM, particularly for $n = 20$ and $n = 30$. However, for $n = 40$, the difference between the two models becomes slightly less pronounced. In the case of the Salinas dataset, the results show more varied outcomes, with inconclusiveness for several Z-values. Nonetheless, the overall trend suggests the superiority of the SPPF model over PSO-FCM, especially for $n = 20$ and $n = 30$. For the Pavia Center dataset, the results from the McNemar's test demonstrate a high variability between SPPF and the PSO-FCM algorithm. The findings are not surprising when considering

the results shown in Figure 5.17, and the overlapping of standard deviations, as it is evident that SPPF and PSO-FCM alternate in their performance superiority across different runs. For all 90 trial runs, SPPF significantly outperforms PSO-FCM in 56 runs, is significantly outperformed in 7 runs, and in 30 runs, there is no statistically significant difference between the two.

Dataset	n	Z_0	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8	Z_9
Indian Pines	20	5.08	13.99	6.93	10.75	3.26	10.07	14.96	14.0	6.79	8.59
	30	9.01	10.67	7.62	5.27	6.56	6.62	8.73	7.86	10.67	4.94
	40	10.12	-0.80	0.98	12.76	3.17	6.23	10.47	7.03	6.38	9.75
Salinas	20	4.68	2.51	4.47	1.04	0.81	1.52	2.18	6.49	0.82	3.91
	30	5.67	-1.18	-1.27	2.56	-1.41	7.89	2.01	3.67	3.69	1.85
	40	-2.57	0.41	3.74	-5.36	-0.37	2.70	-0.92	4.05	-0.66	2.96
Pavia Center	20	1.89	0.40	4.59	3.55	0.42	3.47	-0.93	-3.26	-3.09	4.58
	30	-0.27	5.59	0.25	0.01	-1.35	3.53	0.67	3.37	12.87	0.72
	40	7.22	-2.24	-3.62	8.17	-6.33	-1.58	0.71	3.88	-0.62	2.28

Table 5.16: Experiment 4. Z-values from McNemar's test when comparing SPPF with PSO-FCM.

Conclusion

The findings from this experiment show that SPPF generally outperforms the performance of PSO-FCM for both AVIRIS datasets. The results for the Pavia Center dataset are more inconclusive, and although SPPF achieved the highest mean OA and AA, it is beaten by PSO-FCM for n-values less than 20 and over 40. The overall findings indicate that adding spatial information to the BS process may have advantages that depend on the dataset attributes and characteristics. Comparing results from McNemar's test, SPPF overall shows a slight superiority to PSO-FCM in terms of classification performance.

Comparing the runtime and efficiency of the two models, it is evident that PSO-FCM outperforms SPPF. Such findings were partially expected due to the design choices from the previous experiment (Section 5.4.3), where classification accuracy was prioritized over efficiency.

5.4.5 Experiment 5 - Comparison with Baseline

This experiment investigates how the band subsets selected by the SPPF model compare to the baseline of using all available bands across different hyperspectral datasets. For a fair comparison, the results of SPPF are mainly compared to the baseline of all bands after the removal of bands covering the water absorption region. The complete experiment specifications can be seen in Table 5.17.

Experiment Specifications	
Model	SPPF, Baseline w. noisy band removed
Dataset	Indian Pines, Salinas, Pavia Center
Number of bands selected (n)	5, 10, 15, 20, 25, 30, 35, 40, 45, 50
Runs	20
Stopping criterion	100 generations OR 15 generations without improvement
Number of classification runs	10

Table 5.17: Experiment 5 - Specifications.

(a) Overall accuracy   (b) Average accuracy
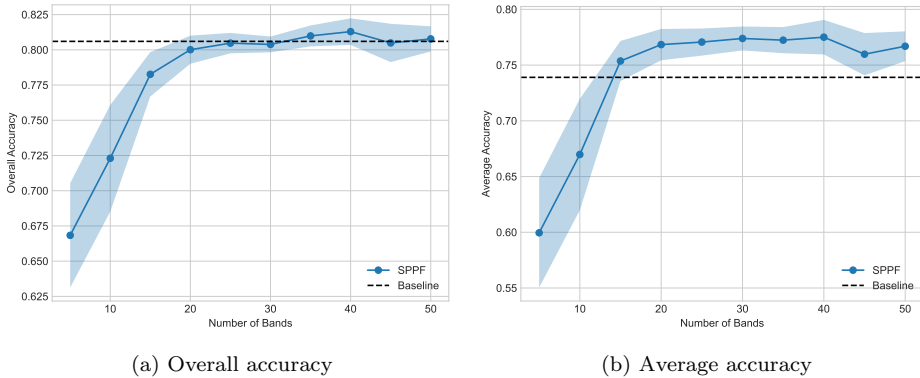
Figure 5.21: Experiment 5 - Performance on Indian Pines.

Focusing initially on the performance of SPPF on the Indian Pines dataset, presented in Figure 5.21, it is observed that SPPF reaches the same overall accuracy as the baseline with around 25 selected bands. However, there is no significant increase in performance with the selection of more bands, possibly due to the "curse of dimensionality." Looking at the average accuracy in (b), it can be seen that SPPF can select band subsets that outperform the baseline AA when the number of selected bands is at least 15. For $n \geq 15$, SPPF appears to

achieve significantly better performance than the baseline with respect to average accuracy. The reason for this improvement in AA can be inferred from looking at Table 5.18, which shows the accuracies for each class for a single classification run with all bands (Baseline) and 30 bands selected by SPPF. Comparing accuracies for classes "Alfalfa" and "Oats" we can see clearly that the bands selected by SPPF are far superior to using all bands when the number of available samples is low, increasing the accuracy by 21.9 and 33.3 percentage points, respectively. This explains why the AA is significantly better for SPPF, yet the OA is about equal to the baseline.

Class	Samples	Baseline (%)	SPPF (%)
Alfalfa	46	41.5	**63.4**
Corn-notill	1428	75.7	**76.2**
Corn-mintill	830	65.7	**66.4**
Corn	237	**68.1**	59.6
Grass-pasture	483	92.4	**93.6**
Grass-trees	730	95.3	**99.3**
Grass-pasture-mowed	28	80.0	**84.0**
Hay-windrowed	478	96.1	**96.5**
Oats	20	55.6	**88.9**
Soybean-notill	972	73.2	**77.6**
Soybean-mintill	2455	**80.3**	77.4
Soybean-clean	593	**74.7**	69.8
Wheat	205	95.7	**96.7**
Woods	1265	**92.6**	92.4
Buildings-Grass-Trees-Drives	386	53.9	**57.1**
Stone-Steel-Towers	93	80.7	**91.2**

Table 5.18: Experiment 5. Accuracy for each individual class on the Indian Pines dataset.

When comparing SPPF with the baseline for the Salinas dataset, displayed in Figure 5.22, it becomes apparent that the performance trends of SPPF mirror those observed with the Indian Pines dataset. Both overall and average accuracy match the baseline performance when 25-30 bands are selected; beyond this, SPPF does not present significant improvements over the baseline in terms of performance. Interestingly, the results of SPPF from the Salinas dataset do not exhibit the same behavior of having a superior AA, as observed on Indian Pines. This is likely explained by the number of samples available for each class being much more evenly distributed for the Salinas dataset, as can be confirmed by looking at Appendix A2.

The similarity of results from both AVIRIS datasets (Indian Pines & Salinas) indicates that the SPPF model is able to maintain the same classification performance whilst reducing the dimensionality of the HSI data. However, the results also show that no significant improvement in classification accuracy is achieved.


(a) Overall accuracy   (b) Average accuracy
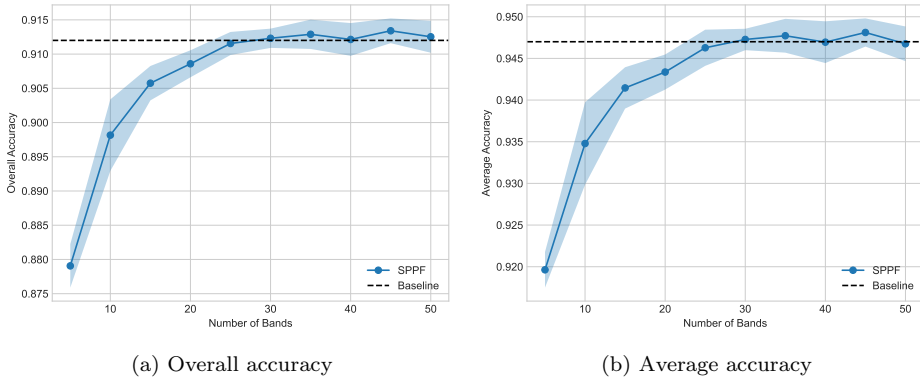
Figure 5.22: Experiment 5 - Performance on Salinas.

The results from the Pavia Center dataset, shown in Figure 5.23, paint a slightly different picture of SPPF's capabilities. Here, SPPF outperforms the baseline in terms of overall accuracy with a minimum of 15 selected bands, while for average accuracy, it takes at least 20 bands. This superiority of SPPF implies that the model is, in some instances, able to identify a subset of bands that reduce dimensionality and improve the classification performance. As discussed in Section 5.4.3, the classification performance decreases when the number of selected bands exceeds 30. However, Figure 5.23 shows that it remains on par or exceeds the performance of the baseline also for higher n-values

Figure 5.24 compares the accuracies for each class between a single run of SPPF with 20 selected bands and the baseline with all bands selected, on the Pavia Center dataset. It can be observed that SPPF's most significant improvements are made in the classification of the "Asphalt" and "Bricks" classes. The baseline using all bands struggles with these classes the most, as seen in (a), so it is no surprise that this is where the most improvements can be made. The accuracies for the other classes are relatively similar between the two classification runs. The figure also shows that the baseline particularly struggles to separate the classes "Trees" and "Asphalt", while the SPPF model does a much better job at this. The reason for this can be observed by looking at Figure 5.25, which shows the mean spectral signatures of the two classes, along with their standard deviation (shaded regions). This figure shows how most of the bands chosen by

(a) Overall accuracy   (b) Average accuracy
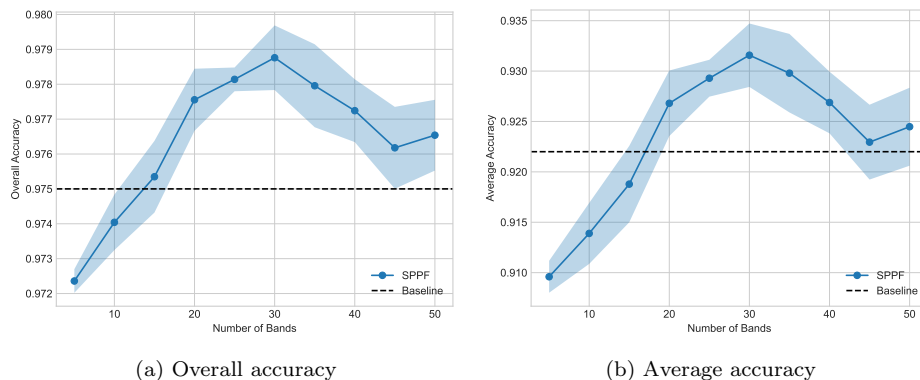
Figure 5.23: Experiment 5 - Performance on Pavia Center.

SPPF (dotted vertical lines) lie in the areas where there is a significant difference in measured spectral intensity between the two classes (illustrated by the divergence of the orange and blue-shaded regions). This is opposed to the baseline, which uses all bands, including the areas of great overlap between the two classes, such as from bands 1 to 20.

Conclusion

This experiment's results highlight the SPPF model's effectiveness in HSI classification tasks. The findings reveal that SPPF, using a notably smaller number of selected bands, can match or even outperform a baseline method that employs all available bands. For the AVIRIS datasets, the results showed that SPPF achieves comparable performance to the baseline with approximately 25 bands. An analysis of per-class accuracy indicated that SPPF has the edge over the baseline in classifying classes with few labeled samples. Specifically, the "Alfalfa" and "Oats" classes in the Indian Pines dataset experienced a significant increase in accuracy, contributing to the superior average accuracy achieved by SPPF.

Interestingly, SPPF's performance on the Pavia Center dataset deviated from the pattern seen in the AVIRIS datasets. Here, SPPF was able to significantly outperform the baseline, possibly being attributed to being able to separate ground truth classes with very similar spectral signatures overall. These findings suggest that SPPF can enhance classification performance under certain conditions while reducing dimensionality.

(a) Baseline   (b) SPPF
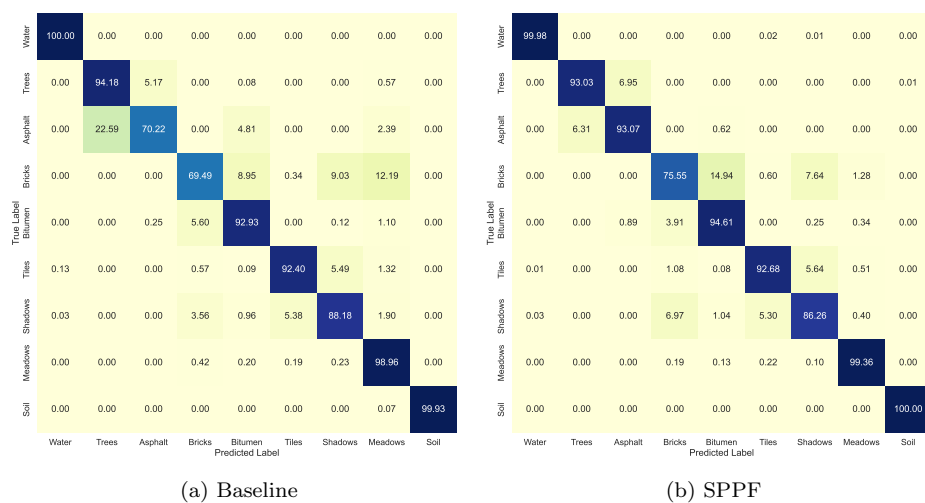
Figure 5.24: Experiment 5 - Confusion matrix from the Pavia Center dataset.
(a) shows the classification performance of a run with all bands (Baseline) and
(b) shows the classification performance with 20 bands selected by SPPF.
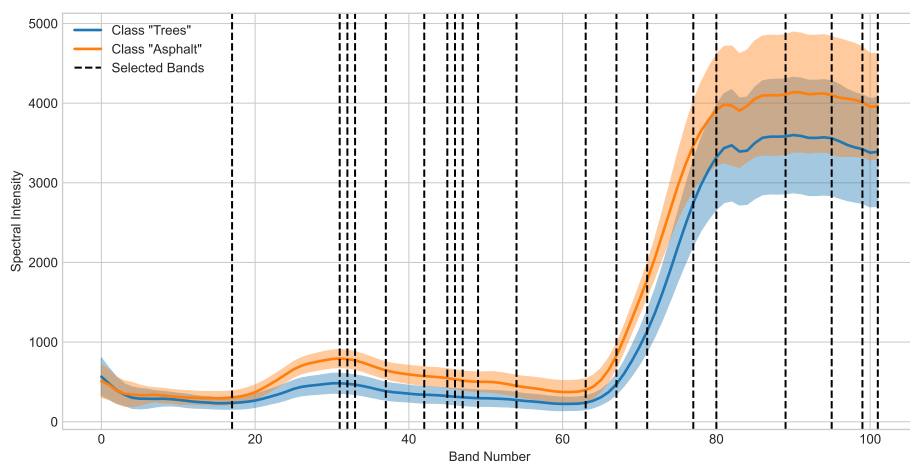
Figure 5.25: Experiment 5. Spectral intensities of classes "Trees" and "Asphalt"
for the Pavia Center dataset.

Chapter 6

Evaluation and Conclusion

This chapter evaluates and discusses the research goal and research questions in Section 6.1. The contributions and limitations are summarized in Section 6.2 and Section 6.3. Their potential implications for future research are elaborated in Section 6.4.

6.1 Evaluation & Discussion

The experiments conducted in Chapter 5 were guided by the research questions and objectives outlined in Chapter 1. This section will provide an assessment of each experiment in relation to the research questions and overall objective.

Research Question 1 *How does pre-removal of noisy bands influence the achieved classification accuracy of unsupervised BS with the superpixel-enhanced PSO-FCM algorithm?*

Based on the experiment results (Section 5.4.1), it was suggested that the SPPF model might be adversely impacted by noisy bands in the datasets, especially when a higher number of bands was selected. It appears that SPPF may form clusters that require the selection of a band with noise. In other words, when a higher number of bands are selected, it potentially leads to the creation of clusters that only consist of low entropy bands, as the experiment conducted by SPPF utilized the highest entropy band criterion for selecting cluster representatives.

Research Question 2 *How can PSO-FCM utilize superpixel information and different divergence measures in order to achieve the best classification accuracy?*

To answer Research Question 2, the FCM objective function was modified to include superpixel information obtained from segmentation, with the goal of enhancing classification accuracy. Four distance metrics incorporating superpixel information have been designed, implemented, and evaluated to answer the question. These distance metrics include SP Euclidean distance, SP KL-Divergence, SP Disjoint Information, and SP KL Divergence-L1Norm and are described in Section 4.2.3.

In addition, four methods for selecting cluster representatives, including Highest entropy, Central tendency, Hybrid weighted-sum criterion, and Hybrid ranking criterion, were proposed and tested to determine the optimal approach for each distance measure. The highest entropy was employed as it was proposed in the original PSO-FCM [Zhang et al., 2017], while Central tendency, Hybrid weighted-sum criterion, and Hybrid ranking criterion were designed for SPPF as reasoned for in Section 4.3.

Experiments on each dataset revealed that solely prioritizing bands with the highest entropy did not consistently lead to improved accuracy when only a few bands were selected. The experimental results of Experiment 2 revealed that selecting the highest entropy band from each cluster was among the worst selection techniques across all datasets and distance metrics, except for the SP Euclidean metric on the Pavia Center dataset. This is likely explained by the Pavia Center dataset's unique entropy profile, with less overall variation and a steady increase in entropy as the band index increases. Overall the results revealed that the Central tendency and the Weighted-sum hybrid selection methods achieved the best performance and were the most consistent.

Furthermore, the experiments revealed that the SP KL-Divergence L1Norm distance metric, in conjunction with the Weighted-sum criterion for cluster representative selection, demonstrated superior performance across multiple datasets, especially when a sufficient number of bands were selected. This finding suggests that considering the relative entropy within each superpixel for each band, rather than relying solely on mean values, contributes to a more nuanced representation of spectral and spatial information.

The utilization of mean superpixel values with divergence measures, specifically SP KL-Divergence and Disjoint Information, yielded less favorable performance in the BS process. Despite the potential of these techniques for enhancing computational efficiency, the classification accuracy was not competitive with the other distance metrics.

Research Question 3 *What is the influence of different distance measures incorporating superpixel information on computational efficiency?*

Experiment 3 (Subsection 5.4.3) revealed a promising improvement in computational efficiency by employing distance measures that operate on aggregated

superpixel mean values rather than individual pixel values. Specifically, employing the SP Euclidean distance metric reduced the number of distance calculations required for SPPF. The benefits of this reduction were also observed in information-based distance measures, as the reduced number of values utilized to compute the histograms contributed to the overall efficiency of the algorithm.

On the other hand, the SP KL-Divergence L1-Norm measure showed a significantly worse runtime, as the overhead calculations of SP KL-Divergence L1-Norm require notably more processing time compared to the other distance metrics. This behavior was expected due to the inherently complex characteristic of this distance metric. The precomputed tensor's size is directly proportional to the number of superpixels, implicating that more superpixels result in a larger tensor and increase the computational overhead (Section 4.2.3).

The comparison of runtimes for the different datasets, shown in Figure 5.13 and Figure 5.14, revealed that the optimization time itself was most heavily dependent on the number of total bands available in the HSI. At the same time, the upfront computational costs most heavily depended on the number of pixels available in the image and the number of superpixels employed in the segmentation process. In general, it was concluded that although superpixels do have the potential to improve the computational efficiency of the BS model, it strongly depends on how the superpixel information is employed within the model.

Research Question 4 *How does the performance and efficiency of the superpixel-enhanced PSO-FCM algorithm compare to the baseline and other state-of-the-art BS methods?*

Based on the experimental results, it was found that the SPPF model is capable of achieving superior performance to the PSO-FCM model, particularly when a sufficient number of bands are chosen. However, experimental results also indicate that this question has slightly different answers depending on the dataset. The performance of the models varied depending on the datasets employed, with SPPF mainly being superior for both the Indian Pines and Salinas datasets. For the Pavia Center dataset, the PSO-FCM model displayed a slight performance advantage. For this dataset, the performance of the SPPF model was hindered by the missing regions of the image due to challenges faced in the superpixel segmentation.

In terms of runtime comparisons, the PSO-FCM model exhibited low variations in runtime across datasets, while the SPPF model's total runtime was largely affected by factors like the number of superpixels and pixels in the image. The experimental results revealed that overall, the efficiency of SPPF is inferior to PSO-FCM, mainly due to the choice of employing a more complex distance metric which requires a higher upfront computational cost. The difference is most pronounced for the Pavia Center dataset, which has a larger pixel count. Despite

the trade-off in efficiency, the SPPF model offers improved classification accuracy by integrating superpixel information.

However, it is important to note that the evaluation of SPPF was limited to comparing it with only one state-of-the-art method, PSO-FCM. Due to time constraints, the proposed method was not compared to other techniques, which could have provided a more comprehensive assessment of its performance and increased confidence in the results.

Goal *To improve classification accuracy and efficiency on HSIs through unsupervised band selection by applying a superpixel-enhanced PSO-FCM hybrid algorithm.*

The goal of this research was partly fulfilled by proposing the superpixel-enhanced PSO-FCM hybrid algorithm, known as SPPF. Across all applied datasets, SPPF demonstrates an overall competitive performance to the classification baseline using all bands. Through analysis of experimental results, it was revealed that SPPF has particular advantages in certain situations, such as classifying classes with few labeled samples and distinguishing classes with very similar spectral signatures. Through these advantages, SPPF was in some cases able to select a band subset that showed a significantly superior classification accuracy compared to the baseline.

Regarding efficiency, it is important to note that PSO-FCM demonstrated better performance compared to SPPF. This outcome aligns with the design choice made in a previous experiment, which prioritized classification accuracy over efficiency by selecting the SP-KL-Divergence-L1Norm distance measure instead of SP Euclidean which showed superior efficiency.

6.2   Contributions

The main contribution of this thesis is the introduction of the SPPF model, a superpixel-enhanced hybrid algorithm for selecting bands in HSIs. This research builds upon the existing application of particle swarm optimization for BS, demonstrating its successful integration with superpixel information. The novelty of this approach lies in the extension of PSO-FCM [Zhang et al., 2017], which incorporates superpixel information into distance measures and divergence measures to offer a unique BS method. Additionally, the thesis proposes combining the distance measures with different cluster-representative selection techniques.

A BS framework has been developed, that allows for easy integration of various superpixel segmentation techniques, distance measures, and representative cluster techniques. The bands selected in this study are provided in the Appendix (A1) for the purpose of reproducibility and future comparison.

6.3 Limitations

The analysis of experimental results for the Pavia Center dataset revealed that the missing image regions proved a challenge for the superpixel segmentation process of SPPF. This provided insights into how the SPPF model handles such situations but could be argued is not ideal for the comparison with PSO-FCM. As a potential alternative, the Pavia University dataset, which shares the same sensor and a similar scene, could have been a better option for the experiments conducted. In the Pavia University dataset, the part of the image that needs to be discarded before analysis is located on the edge of the image, rather than in the middle as observed in the Pavia Center dataset. In such a case, the missing information could be discarded without adversely affecting the superpixel segmentation.

A notable limitation of the thesis pertains to the comparative analysis of the SPPF algorithm. While SPPF was compared to its precursor, PSO-FCM, an in-depth comparison with other state-of-the-art BS methods that incorporate superpixel information was not conducted. For instance, approaches such as SSGA [Zhao et al., 2021] discussed in Chapter 3, have been applied to integrate spatial information in BS. These methods could have served as valuable benchmarks to assess the improvements achieved by SPPF, and to evaluate further whether PSO-FCM was indeed a suitable choice for enhancement with spatial information.

Another limitation in this study comes from understanding the PSO-FCM algorithm, which is expanded upon in the thesis. The original PSO-FCM paper [Zhang et al., 2017] left out many implementation details, so there is a risk that parts of the model might have been misinterpreted. Efforts were made to get additional information from the original authors, but unfortunately, these were unsuccessful. This factor could potentially affect the reliability of the comparative analysis between SPPF and PSO-FCM and should be considered while reviewing the findings.

6.4 Future Work

There are some limitations that can be opportunities for future work. Without further comparisons, the findings are not as comprehensive, and it is difficult to conclusively assess the relative performance of SPPF in the broader context of BS algorithms that incorporate spatial information. Future research should therefore include a more expansive array of comparative methods to evaluate the advantages and limitations of the SPPF model fully.

Furthermore, it would be interesting to investigate the performance of SPPF using different classifiers, particularly ones that employ spatial information in the classification process. Initial exploratory experimentation revealed that such

classifiers have the potential to increase classification accuracy further. Previous work has demonstrated the benefit of directly integrating superpixel information into classification tasks [Wang et al., 2020]. Therefore, investigating whether the simultaneous integration of spatial information in BS and classification grants further benefits or merely introduces redundancy could yield valuable insights.

Besides addressing prevailing limitations, it is possible to explore new research avenues. The literature search revealed that no publicly available superpixel segmentation algorithm has been developed specifically for HSIs, but rather existing algorithms for natural images have been adjusted. Subudhi et al. [2021] also supports this claim. Given the unique features and rich information content of HSIs, it would be interesting to develop a dedicated superpixel segmentation algorithm that considers the spectral information, spatial coherence, and inter-band relationships specific to HSIs. This is an important research direction for the future, as such algorithms could improve BS techniques for HSIs and ultimately enhance the overall analysis and understanding of HSI data.

Bibliography

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and SÃ¼sstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282.

Baisantry, M., Sao, A. K., and Shukla, D. P. (2022). Discriminative spectral-spatial feature extraction-based band selection for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14.

Beheshti, Z. (2019). A time-varying mirrored s-shaped transfer function for binary particle swarm optimization. *Information Sciences*, 512.

Bezdek, J. C., Ehrlich, R., and Full, W. (1984). Fcm: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3):191–203.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3).

Conese, C. and Maselli, F. (1993). Selection of optimum bands from tm scenes through mutual information analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 48(3):2–11.

Dhanachandra, N. and Chanu, Y. (2020). An image segmentation approach based on fuzzy c-means and dynamic particle swarm optimization algorithm. *Multimedia Tools and Applications*, 79:18839–18858.

Di, S., Liao, M., Zhao, Y., Li, Y., and Zeng, Y. (2021). Image superpixel segmentation based on hierarchical multi-level li-slic. *Optics & Laser Technology*, 135:106703.

Duan, Y., Huang, H., and Tang, Y. (2021). Local constraint-based sparse manifold hypergraph learning for dimensionality reduction of hyperspectral image. *IEEE Transactions on Geoscience and Remote Sensing*, 59(1):613–628.

Dunn, J. C. (1973). A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3):32–57.

Fan, Y.-R. and Huang, T.-Z. (2021). Hyperspectral image restoration via super-pixel segmentation of smooth band. *Neurocomputing*, 455:340–352.

Foody, G. M. (2004). Thematic map comparison: Evaluating the statistical significance of differences in classification accuracy. *Photogrammetric engineering and remote sensing*, 70(5):627–634.

Ghamisi, P. and Benediktsson, J. A. (2015). Feature selection based on hybridization of genetic algorithm and particle swarm optimization. *IEEE Geoscience and Remote Sensing Letters*, 12(2):309–313.

Goetz, A. (2009). Three decades of hyperspectral remote sensing of the earth: A personal view. *Remote Sensing of Environment*, 113.

Gong, M., Cai, Q., Chen, X., and Ma, L. (2014). Complex network clustering by multiobjective discrete particle swarm optimization based on decomposition. *IEEE Transactions on Evolutionary Computation*, 18(1):82–97.

Grãna, M., Veganzons, M., and Ayerdi, B. (2014). Hyperspectral remote sensing scenes. `https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes`. Accessed: 2023-02-23.

Grimstad, E. (2019). Evolutionary multi-objective optimization for band selection of hyperspectral imagery using a cluster-based representation. Master's thesis, NTNU.

Harsanyi, J. and Chang, C.-I. (1994). Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach. *IEEE Transactions on Geoscience and Remote Sensing*, 32(4):779–785.

Holland, J. H. (1975). Adaptation in natural and artificial systems. University of Michigan Press. 2nd edition, MIT Press, 1992.

Hossain, M. A., Jia, X., and Pickering, M. (2012). Improved feature selection based on a mutual information measure for hyperspectral image classification. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 3058–3061.

Hsu, C.-W., Chang, C.-C., and Lin, C.-J. (2003). A practical guide to support vector classification. Technical report, Department of Computer Science and Information Engineering, University of National Taiwan, Taipei, Taiwan.

Hughes, G. (1968). On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, 14(1):55–63.

Ji, L., Wang, L., and Geng, X. (2019). An automatic bad band pre-removal method for hyperspectral imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(12):4985–4994.

Jia, S., Yuan, Y., Li, N., Liao, J., Huang, Q., Jia, X., and Xu, M. (2022). A multiscale superpixel-level group clustering framework for hyperspectral band selection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–18.

Jia, S., Zhan, Z., Zhang, M., Xu, M., Huang, Q., Zhou, J., and Jia, X. (2021). Multiple feature-based superpixel-level decision fusion for hyperspectral and lidar data classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(2):1437–1452.

Kar, D. A., Patro, R. N., Subudhi, S., and Biswal, P. K. (2019). Histogram based automatic noisy band removal for remotely sensed hyperspectral images. In *3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 1237–1242.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4.

Kennedy, J. and Eberhart, R. (1997). A discrete binary version of the particle swarm algorithm. In *IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 5, pages 4104–4108.

Kullback, S. and Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86.

Kumar, S. (2022). C-means clustering explained. `https://builtin.com/data-science/c-means`. Accessed: 2023-03-23.

Liu, M.-Y., Tuzel, O., Ramalingam, S., and Chellappa, R. (2011). Entropy rate superpixel segmentation. In *CVPR 2011*, pages 2097–2104.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press.

Martinez-Uso, A., Pla, F., Sotoca, J. M., and Garcia-Sevilla, P. (2007). Clustering-based hyperspectral band selection using information measures. *IEEE Transactions on Geoscience and Remote Sensing*, 45(12):4158–4171.

Mirjalili, S. M. and Lewis, A. (2013). S-shaped versus v-shaped transfer functions for binary particle swarm optimization. *Swarm Evol. Comput.*, 9:1–14.

Muller-Karger, F., Roffer, M., Walker, N., Oliver, M., Schofield, O., Abbott, M., Graber, H., Leben, R., and Goni, G. (2013). Satellite remote sensing in support of an integrated ocean observing system. *IEEE Geoscience and Remote Sensing Magazine*, 1(4):8–18.

Optics, E. (2022). Hyperspectral and multispectral imaging: Edmund optics. `https://www.edmundoptics.com/knowledge-center/application-notes/imaging/hyperspectral-and-multispectral-imaging/`. Accessed: 2023-03-23.

Patro, R. N., Subudhi, S., Biswal, P. K., and Dellacqua, F. (2021). A review of unsupervised band selection techniques: Land cover classification for hyperspectral earth observation data. *IEEE Geoscience and Remote Sensing Magazine*, 9(3):72–111.

Paul, A. and Chaki, N. (2022). Band selection using spectral and spatial information in particle swarm optimization for hyperspectral image classification. *Soft Computing*, 26(13).

Petrie, G., Heasler, P., and Warner, T. (1998). Optimal band selection strategies for hyperspectral data sets. In *IGARSS '98. Sensing and Managing the Environment. 1998 IEEE International Geoscience and Remote Sensing. Symposium Proceedings*, volume 3, pages 1582–1584.

Ren and Malik (2003). Learning a classification model for segmentation. In *Proceedings Ninth IEEE International Conference on Computer Vision*, volume 1, pages 10–17.

Ren, L., Zhao, L., and Wang, Y. (2019). A superpixel-based dual window rx for hyperspectral anomaly detection. *IEEE Geoscience and Remote Sensing Letters*, 17(7):1233–1237.

Shami, T. M., El-Saleh, A. A., Alswaitti, M., Al-Tashi, Q., Summakieh, M. A., and Mirjalili, S. (2022). Particle swarm optimization: A comprehensive survey. *IEEE Access*, 10:10031–10061.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423.

Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

Stutz, D., Hermans, A., and Leibe, B. (2018). Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding*, 166:1–27.

Su, H., Du, Q., Chen, G., and Du, P. (2014). Optimized hyperspectral band selection using particle swarm optimization. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6):2659–2670.

Subudhi, S., Patro, R. N., Biswal, P. K., and Dell'Acqua, F. (2021). A survey on superpixel segmentation as a preprocessing step in hyperspectral image analysis. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:5015–5035.

Sun, L., Jeon, B., Soomro, B. N., Zheng, Y., Wu, Z., and Xiao, L. (2018). Fast superpixel based subspace low rank learning method for hyperspectral denoising. *IEEE Access*, 6:12031–12043.

Sun, W. and Du, Q. (2019). Hyperspectral band selection: A review. *IEEE Geoscience and Remote Sensing Magazine*, 7(2):118–139.

Tan, Y., Lu, L., Bruzzone, L., Guan, R., Chang, Z., and Yang, C. (2020). Hyperspectral band selection for lithologic discrimination and geological mapping. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:471–486.

Tschannerl, J., Ren, J., Yuen, P., Sun, G., Zhao, H., Yang, Z., Wang, Z., and Marshall, S. (2019). Mimr-dgsa: Unsupervised hyperspectral band selection based on information theory and a modified discrete gravitational search algorithm. *Information Fusion*, 51:189–200.

Verma, H., Verma, D., and Tiwari, P. K. (2021). A population based hybrid fcm-pso algorithm for clustering analysis and segmentation of brain image. *Expert Systems with Applications*, 167:114121.

Wang, C., Gong, M., Zhang, M., and Chan, Y. (2015). Unsupervised hyperspectral image band selection via column subset selection. *IEEE Geoscience and Remote Sensing Letters*, 12(7):1411–1415.

Wang, J. and Chang, C. I. (2006). Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 44(6):1586–1600.

Wang, X., Zhong, Y., Zhang, L., and Xu, Y. (2017). Spatial group sparsity regularized nonnegative matrix factorization for hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 55(11):6287–6304.

Wang, Y., Yu, W., and Fang, Z. (2020). Multiple kernel-based svm classification of hyperspectral images by combining spectral, spatial, and semantic information. *Remote Sensing*, 12(1):120.

Yang, C., Bruzzone, L., Zhao, H., Tan, Y., and Guan, R. (2018). Superpixel-based unsupervised band selection for classification of hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing*, 56(12):7230–7245.

Yang, R., Su, L., Zhao, X., Wan, H., and Sun, J. (2017). Representative band selection for hyperspectral image classification. *Journal of Visual Communication and Image Representation*, 48:396–403.

Zhang, M., Ma, J., and Gong, M. (2017). Unsupervised hyperspectral band selection by fuzzy clustering with particle swarm optimization. *IEEE Geoscience and Remote Sensing Letters*, 14(5):773–777.

Zhang, W., Li, X., and Zhao, L. (2018). A fast hyperspectral feature selection method based on band correlation analysis. *IEEE Geoscience and Remote Sensing Letters*, 15(11):1750–1754.

Zhang, Y., Desai, M. D., Zhang, J., and Jin, M. (1999). Adaptive subspace decomposition for hyperspectral data dimensionality reduction. In *Proceedings of the 1999 International Conference on Image Processing*, pages 326–329.

Zhao, H., Bruzzone, L., Guan, R., Zhou, F., and Yang, C. (2021). Spectral-spatial genetic algorithm-based unsupervised band selection for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(11):9616–9632.

Zheng, X., Yuan, Y., and Lu, X. (2017). Dimensionality reduction by spatial-spectral preservation in selected bands. *IEEE Transactions on Geoscience and Remote Sensing*, 55(9):5185–5197.

Appendices

A1   Experimental Runs

n	Trial	SPPF	PSO-FCM
20	Z_0	[12, 31, 26, 51, 97, 62, 82, 80, 101, 114, 116, 122, 124, 127, 128, 130, 132, 134, 136, 180]	[33, 30, 29, 36, 41, 46, 39, 45, 53, 59, 74, 84, 96, 107, 114, 126, 112, 106, 111, 157]
20	Z_1	[25, 11, 43, 46, 55, 5, 64, 62, 81, 85, 179, 114, 115, 116, 118, 131, 99, 129, 135, 101]	[0, 37, 39, 41, 46, 53, 33, 45, 16, 74, 30, 60, 59, 115, 121, 112, 106, 107, 111, 197]
20	Z_2	[9, 24, 36, 63, 47, 71, 84, 96, 81, 80, 79, 114, 115, 117, 120, 124, 126, 128, 132, 109]	[45, 41, 50, 46, 39, 17, 53, 33, 30, 74, 59, 57, 106, 114, 116, 129, 112, 194, 111, 107]
20	Z_3	[12, 26, 30, 42, 52, 55, 5, 64, 62, 81, 79, 114, 116, 118, 120, 125, 126, 130, 136, 179]	[39, 41, 46, 53, 33, 45, 69, 74, 83, 30, 96, 59, 114, 124, 112, 107, 106, 111, 108, 191]
20	Z_4	[9, 21, 26, 36, 47, 58, 59, 67, 73, 80, 82, 81, 114, 115, 117, 123, 124, 130, 132, 141]	[16, 33, 25, 29, 30, 34, 41, 47, 46, 45, 68, 39, 112, 116, 126, 106, 152, 111, 107, 155]
20	Z_5	[7, 11, 24, 27, 35, 46, 55, 79, 82, 81, 80, 106, 133, 114, 116, 117, 118, 124, 128, 167]	[41, 48, 46, 33, 45, 53, 39, 59, 30, 60, 31, 74, 111, 112, 116, 126, 194, 168, 107, 106]
20	Z_6	[24, 9, 36, 47, 58, 69, 82, 80, 137, 115, 117, 120, 123, 124, 127, 130, 132, 135, 164, 185]	[33, 41, 46, 47, 39, 45, 53, 56, 59, 74, 30, 87, 29, 113, 124, 112, 106, 107, 111, 151]
20	Z_7	[12, 26, 42, 46, 51, 69, 73, 64, 35, 79, 80, 81, 82, 178, 115, 117, 124, 123, 129, 134]	[39, 41, 46, 48, 33, 54, 45, 53, 75, 30, 27, 59, 112, 115, 114, 121, 194, 107, 111, 106]
20	Z_8	[20, 26, 38, 40, 52, 73, 63, 69, 2, 81, 78, 112, 116, 122, 125, 130, 132, 133, 134, 109]	[41, 46, 47, 59, 33, 45, 53, 54, 30, 27, 74, 29, 111, 113, 123, 106, 110, 108, 107]
20	Z_9	[9, 12, 25, 46, 47, 73, 64, 69, 71, 82, 59, 113, 114, 115, 117, 120, 125, 130, 135, 179]	[41, 46, 48, 56, 68, 53, 70, 33, 75, 87, 59, 30, 107, 115, 121, 112, 106, 155, 111, 157]
30	Z_0	[5, 9, 33, 21, 22, 23, 26, 55, 58, 44, 69, 70, 61, 76, 89, 83, 80, 100, 110, 111, 115, 117, 122, 124, 129, 132, 135, 137, 158, 184]	[15, 18, 20, 30, 33, 39, 41, 48, 47, 46, 59, 54, 45, 66, 53, 137, 75, 84, 86, 60, 74, 112, 115, 119, 121, 106, 111, 107, 155, 191]
30	Z_1	[12, 26, 46, 43, 52, 69, 73, 58, 59, 63, 35, 89, 81, 84, 95, 80, 174, 113, 114, 116, 122, 123, 126, 129, 78, 100, 135, 101, 139, 191]	[39, 41, 48, 46, 54, 55, 33, 45, 53, 71, 72, 73, 75, 74, 25, 60, 30, 59, 115, 114, 120, 123, 112, 111, 151, 110, 107, 155, 106, 191]
30	Z_2	[9, 19, 25, 26, 31, 3, 42, 52, 70, 58, 73, 63, 66, 69, 89, 80, 85, 94, 99, 113, 115, 117, 118, 120, 124, 100, 136, 139, 183, 163]	[0, 37, 41, 47, 46, 53, 39, 59, 15, 45, 67, 69, 33, 75, 74, 30, 20, 97, 96, 107, 115, 114, 112, 146, 151, 111, 123, 106, 110, 108, 107, 190]
30	Z_3	[23, 9, 40, 38, 46, 49, 51, 52, 53, 58, 63, 64, 68, 55, 89, 79, 83, 115, 117, 119, 123, 124, 128, 131, 134, 137, 187, 157, 163, 170]	[8, 30, 15, 40, 41, 42, 46, 48, 56, 59, 67, 53, 54, 39, 83, 84, 75, 60, 29, 106, 112, 116, 125, 119, 111, 108, 107, 155, 185, 151]
30	Z_4	[11, 26, 42, 52, 53, 58, 63, 66, 69, 71, 61, 82, 75, 94, 80, 112, 114, 116, 120, 121, 123, 125, 127, 129, 100, 136, 188, 110, 165, 174]	[33, 25, 29, 30, 42, 41, 46, 48, 44, 65, 69, 53, 39, 75, 74, 83, 59, 106, 115, 114, 121, 112, 111, 146, 158, 109, 110, 168, 107]
30	Z_5	[7, 10, 22, 25, 27, 31, 32, 18, 42, 46, 49, 51, 72, 63, 68, 79, 101, 82, 81, 84, 80, 134, 115, 116, 118, 119, 130, 138, 165, 184]	[33, 37, 41, 42, 48, 46, 45, 66, 67, 53, 70, 39, 74, 60, 57, 59, 30, 106, 114, 115, 124, 113, 111, 158, 110, 157, 107, 152, 151]
30	Z_6	[9, 18, 23, 26, 42, 49, 52, 66, 63, 62, 55, 80, 82, 81, 94, 98, 174, 114, 115, 116, 117, 118, 123, 124, 129, 100, 134, 138, 187, 164]	[36, 43, 42, 41, 48, 46, 59, 33, 45, 69, 53, 39, 75, 74, 83, 84, 85, 30, 97, 104, 110, 114, 122, 116, 126, 113, 111, 155, 107, 106]
30	Z_7	[9, 14, 21, 26, 36, 42, 52, 55, 61, 35, 63, 66, 68, 69, 70, 72, 73, 81, 89, 84, 96, 94, 98, 80, 139, 114, 128, 136, 185, 165]	[37, 39, 41, 42, 46, 48, 53, 54, 63, 15, 69, 33, 74, 83, 20, 87, 81, 30, 29, 59, 112, 114, 116, 119, 111, 107, 152, 168, 110, 106]
30	Z_8	[9, 21, 30, 27, 31, 42, 52, 70, 71, 73, 64, 89, 79, 81, 84, 94, 96, 97, 80, 114, 116, 119, 121, 122, 127, 130, 135, 187, 169, 110]	[0, 20, 24, 30, 29, 33, 41, 46, 48, 59, 56, 45, 69, 53, 54, 39, 87, 74, 75, 83, 113, 115, 119, 111, 107, 110, 169, 109, 106, 185]
30	Z_9	[12, 26, 44, 40, 47, 45, 53, 55, 74, 5, 67, 62, 1, 80, 92, 86, 94, 111, 114, 115, 116, 118, 120, 124, 125, 127, 100, 136, 165, 185]	[0, 36, 39, 41, 48, 50, 46, 54, 16, 45, 67, 53, 71, 33, 59, 30, 29, 107, 115, 114, 123, 129, 113, 111, 106, 168, 110, 108, 194, 197]
40	Z_0	[9, 19, 24, 26, 35, 37, 42, 47, 48, 49, 52, 58, 63, 67, 55, 79, 82, 84, 81, 95, 97, 98, 80, 101, 136, 114, 115, 117, 118, 120, 126, 130, 132, 134, 148, 154, 174, 163, 171, 187]	[34, 33, 20, 29, 30, 36, 46, 48, 41, 50, 52, 39, 56, 73, 70, 45, 67, 65, 53, 75, 83, 84, 60, 87, 59, 196, 106, 112, 115, 116, 114, 121, 113, 134, 111, 151, 160, 110, 108, 107]
40	Z_1	[9, 18, 25, 40, 38, 52, 53, 60, 63, 67, 62, 71, 73, 89, 82, 84, 94, 81, 95, 80, 108, 112, 114, 115, 116, 117, 118, 120, 123, 124, 127, 130, 132, 137, 140, 158, 168, 174, 162, 187]	[6, 8, 16, 17, 29, 30, 33, 37, 41, 42, 48, 46, 53, 45, 67, 69, 54, 39, 75, 74, 59, 84, 83, 57, 107, 108, 110, 113, 115, 114, 121, 126, 112, 111, 152, 106, 168, 157, 155, 191]
40	Z_2	[9, 20, 24, 26, 32, 56, 46, 45, 43, 50, 69, 57, 59, 64, 62, 71, 73, 89, 79, 81, 91, 85, 98, 101, 134, 114, 116, 119, 120, 124, 123, 127, 129, 131, 137, 109, 187, 159, 163, 168]	[16, 33, 21, 30, 29, 31, 34, 7, 37, 41, 46, 47, 48, 59, 65, 66, 53, 54, 55, 39, 74, 60, 81, 75, 76, 194, 107, 115, 114, 121, 126, 113, 112, 111, 106, 110, 108, 167, 168, 155]
40	Z_3	[4, 13, 23, 26, 33, 42, 36, 49, 52, 54, 58, 59, 61, 62, 64, 55, 80, 82, 93, 85, 90, 81, 97, 98, 77, 110, 112, 116, 118, 120, 126, 127, 100, 132, 134, 136, 138, 165, 176, 151]	[20, 24, 29, 30, 33, 37, 41, 48, 50, 46, 53, 39, 45, 69, 70, 71, 84, 60, 91, 75, 59, 74, 57, 104, 107, 111, 113, 115, 119, 123, 121, 128, 112, 139, 106, 110, 168, 194, 145]
40	Z_4	[8, 23, 26, 31, 12, 2, 36, 39, 42, 49, 52, 72, 58, 61, 69, 63, 67, 89, 81, 85, 79, 139, 112, 113, 114, 115, 116, 118, 122, 127, 129, 131, 132, 135, 101, 140, 194, 173, 164, 179]	[8, 16, 17, 29, 30, 33, 34, 41, 44, 46, 48, 50, 56, 63, 67, 53, 54, 39, 77, 57, 74, 87, 91, 75, 59, 96, 58, 106, 111, 112, 114, 116, 126, 133, 137, 153, 109, 110, 107, 151]
40	Z_5	[10, 25, 40, 38, 46, 49, 52, 51, 58, 74, 65, 66, 68, 71, 79, 78, 80, 75, 94, 85, 98, 100, 101, 165, 138, 135, 114, 116, 118, 120, 124, 123, 127, 129, 130, 132, 134, 139, 176, 191]	[0, 17, 18, 20, 30, 29, 31, 33, 37, 38, 39, 41, 42, 46, 45, 48, 65, 69, 53, 54, 74, 84, 85, 59, 57, 105, 106, 115, 114, 120, 128, 113, 112, 151, 158, 111, 110, 107, 168, 155]
40	Z_6	[7, 33, 10, 25, 2, 37, 40, 41, 42, 46, 50, 53, 55, 58, 67, 63, 70, 73, 78, 81, 82, 93, 85, 90, 91, 97, 187, 113, 114, 115, 117, 120, 123, 125, 127, 130, 133, 137, 148, 165]	[0, 33, 25, 30, 36, 37, 41, 47, 48, 46, 39, 74, 56, 62, 45, 53, 71, 75, 83, 87, 59, 84, 96, 97, 57, 115, 117, 114, 122, 113, 111, 150, 151, 165, 110, 168, 107, 106]
40	Z_7	[9, 24, 2, 36, 42, 47, 52, 73, 58, 63, 68, 70, 72, 76, 81, 82, 86, 89, 94, 92, 93, 95, 97, 80, 100, 114, 115, 117, 118, 121, 120, 125, 127, 130, 134, 138, 188, 169, 176, 163]	[0, 8, 16, 15, 30, 34, 35, 36, 41, 44, 46, 43, 48, 56, 65, 67, 68, 39, 57, 75, 83, 84, 60, 92, 59, 103, 151, 113, 115, 117, 119, 121, 112, 195, 152, 107, 109, 111, 168, 106]
40	Z_8	[2, 14, 18, 23, 26, 46, 47, 57, 58, 59, 61, 35, 64, 69, 70, 71, 72, 73, 76, 80, 85, 96, 90, 88, 81, 99, 100, 132, 114, 115, 117, 118, 124, 127, 136, 168, 110, 176, 181, 191]	[0, 11, 16, 30, 34, 37, 42, 41, 48, 46, 39, 69, 45, 68, 53, 72, 73, 56, 76, 88, 87, 59, 82, 83, 57, 113, 119, 123, 116, 112, 111, 140, 195, 146, 152, 106, 107, 110, 157, 147]
40	Z_9	[5, 7, 9, 21, 26, 16, 36, 40, 38, 45, 46, 50, 49, 52, 58, 61, 67, 71, 55, 76, 78, 80, 81, 85, 89, 98, 112, 114, 116, 130, 120, 127, 124, 132, 137, 139, 188, 165, 170, 176]	[0, 8, 9, 12, 16, 33, 20, 30, 41, 42, 48, 46, 53, 39, 45, 69, 70, 71, 83, 84, 74, 87, 75, 85, 96, 59, 195, 115, 114, 128, 129, 112, 111, 141, 152, 157, 110, 108, 107, 106]

Table A1: Selected bands for each trial of McNemars test between SPPF and PSO-FCM on the Indian Pines Dataset.

n	Trial	SPPF	PSO-FCM
20	Z_0	[20, 35, 37, 45, 51, 55, 65, 66, 67, 69, 59, 76, 98, 92, 139, 131, 157, 164, 169, 112]	[36, 31, 34, 47, 54, 55, 56, 59, 75, 61, 78, 100, 198, 120, 79, 190, 157, 80, 159, 197]
20	Z_1	[21, 35, 36, 37, 38, 47, 53, 49, 58, 66, 69, 71, 73, 76, 90, 84, 119, 132, 115, 112]	[7, 36, 35, 34, 47, 55, 68, 57, 59, 85, 63, 61, 198, 159, 120, 79, 80, 157, 158, 190]
20	Z_2	[35, 19, 45, 47, 55, 56, 76, 74, 66, 69, 70, 72, 89, 97, 126, 128, 104, 161, 167, 187]	[12, 15, 35, 34, 46, 51, 55, 57, 59, 85, 63, 61, 201, 190, 81, 79, 197, 157, 80, 159]
20	Z_3	[32, 14, 19, 40, 45, 51, 55, 57, 65, 67, 69, 72, 99, 92, 97, 131, 113, 170, 115]	[16, 15, 35, 34, 46, 51, 55, 56, 59, 85, 63, 61, 120, 79, 197, 159, 157, 80, 183, 201]
20	Z_4	[27, 14, 19, 46, 52, 49, 56, 58, 61, 66, 74, 64, 90, 100, 101, 119, 132, 167, 169, 113]	[80, 36, 31, 34, 47, 51, 55, 56, 59, 61, 85, 77, 97, 197, 120, 79, 161, 157, 159, 183]
20	Z_5	[21, 35, 37, 39, 42, 47, 41, 53, 57, 66, 67, 68, 71, 76, 86, 99, 92, 97, 131, 183]	[36, 27, 31, 34, 46, 51, 59, 61, 85, 58, 96, 190, 120, 79, 80, 198, 152, 157, 161, 159]
20	Z_6	[35, 20, 37, 46, 48, 53, 50, 56, 58, 64, 70, 75, 76, 77, 85, 90, 97, 125, 136, 185]	[15, 36, 35, 34, 46, 47, 55, 56, 59, 85, 63, 61, 197, 81, 79, 198, 157, 80, 159, 185]
20	Z_7	[25, 34, 16, 40, 44, 41, 47, 53, 56, 59, 69, 74, 65, 64, 89, 86, 97, 135, 131, 183]	[36, 35, 34, 47, 55, 59, 56, 74, 61, 96, 63, 198, 187, 120, 79, 80, 157, 158, 159, 190]
20	Z_8	[35, 20, 37, 43, 45, 51, 55, 64, 68, 56, 72, 74, 97, 99, 4, 92, 103, 124, 114]	[80, 37, 26, 31, 34, 51, 55, 59, 70, 56, 63, 61, 96, 198, 190, 120, 79, 158, 157, 159]
20	Z_9	[21, 14, 32, 19, 37, 46, 55, 66, 68, 39, 70, 57, 74, 76, 99, 92, 97, 131, 188, 168]	[38, 15, 36, 24, 34, 47, 51, 57, 61, 59, 96, 80, 120, 79, 197, 157, 161, 156, 159, 201]
30	Z_0	[11, 21, 28, 34, 37, 47, 44, 41, 46, 48, 52, 78, 66, 69, 56, 74, 79, 97, 90, 91, 92, 94, 100, 102, 139, 117, 131, 157, 187, 165]	[36, 24, 35, 31, 34, 46, 49, 51, 55, 67, 69, 56, 59, 85, 63, 96, 61, 116, 120, 117, 122, 79, 185, 197, 157, 80, 158, 159, 190, 198]
30	Z_1	[27, 14, 19, 38, 46, 51, 54, 49, 56, 68, 70, 57, 73, 74, 76, 87, 4, 92, 97, 99, 132, 128, 129, 104, 140, 144, 195, 165, 171, 178]	[5, 12, 36, 30, 31, 34, 40, 44, 47, 52, 55, 70, 57, 59, 75, 96, 63, 61, 85, 118, 120, 125, 79, 80, 201, 185, 157, 171, 159, 197]
30	Z_2	[23, 37, 42, 45, 51, 54, 49, 59, 69, 72, 75, 64, 77, 85, 91, 93, 94, 99, 97, 83, 118, 126, 125, 130, 134, 104, 140, 189, 169, 167]	[12, 15, 26, 34, 31, 47, 52, 55, 67, 56, 59, 75, 96, 61, 63, 197, 185, 120, 128, 79, 133, 81, 115, 80, 159, 157, 190, 198, 201]
30	Z_3	[20, 28, 31, 13, 48, 47, 39, 57, 58, 38, 67, 69, 71, 72, 76, 97, 85, 94, 89, 100, 7, 92, 115, 116, 119, 128, 126, 104, 171, 188]	[36, 25, 31, 34, 35, 47, 51, 55, 59, 77, 69, 56, 74, 76, 63, 62, 96, 61, 197, 120, 126, 133, 81, 185, 199, 157, 80, 156, 190, 201]
30	Z_4	[23, 24, 37, 38, 41, 46, 50, 49, 56, 58, 76, 66, 68, 71, 74, 78, 97, 99, 4, 90, 103, 116, 120, 124, 126, 134, 139, 161, 167, 189]	[11, 15, 37, 36, 31, 34, 46, 51, 52, 55, 69, 56, 59, 60, 77, 61, 96, 85, 197, 159, 189, 120, 117, 133, 199, 161, 157, 80, 170, 190]
30	Z_5	[192, 11, 20, 23, 27, 31, 34, 36, 39, 45, 53, 47, 48, 66, 68, 69, 70, 71, 59, 76, 84, 4, 92, 121, 129, 104, 139, 156, 165, 179]	[0, 80, 37, 36, 35, 34, 46, 56, 49, 55, 59, 70, 72, 74, 87, 63, 96, 61, 85, 198, 190, 120, 117, 79, 143, 157, 156, 178, 158, 159]
30	Z_6	[11, 22, 26, 32, 36, 39, 45, 51, 55, 58, 64, 66, 69, 56, 71, 72, 74, 77, 89, 100, 92, 97, 83, 196, 137, 117, 122, 134, 179, 169]	[11, 15, 36, 31, 34, 42, 44, 45, 47, 55, 69, 56, 57, 59, 85, 96, 63, 61, 201, 183, 118, 127, 79, 81, 80, 197, 159, 157, 179, 190]
30	Z_7	[16, 28, 36, 37, 39, 45, 42, 43, 51, 53, 48, 61, 66, 68, 69, 71, 57, 75, 64, 85, 86, 90, 95, 4, 102, 117, 129, 140, 162, 185]	[36, 23, 25, 34, 31, 47, 50, 51, 55, 59, 76, 69, 56, 85, 61, 63, 96, 79, 80, 116, 120, 126, 145, 201, 158, 185, 157, 156, 159, 197]
30	Z_8	[16, 25, 29, 31, 36, 40, 46, 53, 56, 58, 61, 38, 68, 72, 75, 64, 94, 99, 90, 92, 101, 83, 134, 122, 151, 160, 168, 142, 114, 196]	[8, 36, 21, 35, 27, 34, 43, 46, 47, 51, 55, 56, 59, 62, 85, 89, 63, 61, 201, 197, 159, 190, 116, 120, 79, 80, 157, 166, 169, 158]
30	Z_9	[22, 30, 14, 37, 41, 45, 50, 47, 54, 49, 58, 63, 64, 66, 68, 70, 74, 89, 84, 7, 92, 119, 124, 123, 103, 133, 138, 165, 172, 188]	[9, 15, 36, 34, 31, 44, 46, 51, 52, 55, 68, 56, 57, 59, 85, 61, 63, 84, 79, 199, 120, 126, 198, 183, 159, 157, 161, 156, 190, 197]
40	Z_0	[16, 30, 28, 35, 45, 52, 56, 60, 62, 65, 66, 67, 69, 72, 58, 64, 63, 84, 87, 7, 91, 93, 94, 99, 95, 4, 102, 116, 117, 120, 118, 124, 125, 129, 133, 139, 179, 168, 174, 192]	[80, 14, 15, 37, 36, 26, 31, 34, 44, 47, 51, 54, 55, 67, 69, 57, 59, 60, 81, 63, 84, 85, 61, 104, 109, 117, 120, 133, 116, 198, 199, 158, 157, 161, 156, 172, 176, 159, 190, 201]
40	Z_1	[16, 26, 32, 37, 38, 39, 40, 46, 48, 50, 54, 55, 56, 57, 58, 64, 70, 79, 97, 100, 89, 92, 94, 115, 118, 124, 128, 130, 133, 136, 140, 0, 194, 151, 181, 161, 165, 157, 171, 173]	[36, 21, 25, 36, 31, 43, 46, 47, 49, 51, 55, 57, 59, 76, 71, 74, 63, 85, 93, 94, 61, 88, 197, 117, 118, 120, 79, 81, 139, 80, 201, 199, 185, 190, 157, 163, 156, 159, 183, 198]
40	Z_2	[20, 26, 32, 14, 45, 44, 43, 48, 51, 54, 55, 56, 42, 59, 65, 66, 67, 69, 72, 64, 79, 82, 97, 87, 92, 94, 100, 102, 195, 117, 120, 125, 130, 135, 137, 141, 165, 180, 172, 186]	[80, 36, 27, 32, 31, 34, 38, 44, 47, 49, 51, 52, 55, 59, 68, 71, 74, 60, 61, 85, 63, 97, 94, 201, 116, 120, 117, 127, 183, 189, 199, 159, 157, 161, 178, 176, 190, 197]
40	Z_3	[0, 11, 20, 26, 31, 37, 38, 39, 41, 46, 51, 55, 64, 65, 68, 56, 71, 72, 59, 75, 42, 63, 83, 97, 88, 4, 90, 92, 117, 119, 124, 132, 129, 135, 137, 168, 167, 172, 181, 195]	[11, 15, 36, 23, 27, 29, 31, 34, 40, 45, 46, 47, 54, 55, 76, 69, 59, 86, 78, 63, 94, 61, 85, 98, 183, 119, 126, 79, 133, 81, 80, 198, 185, 158, 163, 162, 157, 159, 190, 200]
40	Z_4	[18, 22, 24, 27, 32, 14, 45, 42, 44, 48, 51, 53, 52, 55, 66, 69, 56, 71, 57, 74, 76, 77, 63, 94, 91, 90, 148, 116, 121, 123, 125, 129, 133, 104, 140, 164, 162, 175, 114, 190]	[10, 15, 36, 24, 34, 31, 35, 43, 46, 47, 54, 55, 57, 59, 77, 67, 70, 73, 86, 85, 92, 93, 61, 96, 107, 185, 115, 120, 126, 79, 133, 81, 80, 197, 190, 158, 159, 162, 157, 203]
40	Z_5	[11, 17, 19, 23, 28, 31, 34, 36, 37, 51, 42, 57, 64, 65, 66, 68, 69, 71, 73, 75, 82, 96, 100, 7, 78, 90, 94, 97, 83, 118, 121, 127, 132, 135, 139, 152, 161, 165, 175, 155]	[12, 15, 21, 24, 31, 34, 35, 36, 44, 46, 51, 55, 54, 67, 56, 57, 59, 61, 78, 63, 96, 85, 201, 197, 183, 117, 118, 120, 125, 79, 116, 81, 80, 152, 161, 157, 169, 176, 159, 190]
40	Z_6	[7, 16, 25, 29, 37, 41, 44, 46, 48, 51, 53, 55, 42, 57, 64, 66, 67, 71, 73, 74, 75, 60, 82, 83, 85, 86, 87, 90, 63, 97, 102, 116, 119, 124, 132, 139, 1, 190, 165, 176]	[7, 36, 22, 26, 31, 34, 39, 46, 49, 51, 55, 65, 68, 56, 57, 59, 77, 63, 96, 85, 89, 93, 61, 197, 119, 123, 126, 79, 80, 198, 196, 183, 156, 157, 170, 176, 159, 190, 203]
40	Z_7	[10, 23, 34, 28, 31, 19, 38, 40, 46, 48, 53, 52, 56, 58, 63, 64, 65, 68, 71, 73, 74, 76, 78, 89, 86, 97, 83, 140, 116, 118, 120, 122, 127, 134, 137, 192, 180, 165, 171, 173]	[102, 28, 31, 33, 34, 42, 44, 46, 47, 51, 55, 77, 66, 56, 59, 76, 79, 85, 78, 63, 94, 61, 96, 201, 190, 116, 120, 126, 133, 134, 114, 161, 157, 163, 158, 159, 187, 197, 198, 199]
40	Z_8	[11, 19, 23, 32, 34, 43, 41, 46, 47, 48, 53, 40, 56, 58, 64, 38, 67, 39, 70, 72, 73, 74, 75, 76, 89, 100, 92, 97, 119, 124, 129, 135, 140, 161, 180, 167, 178, 172, 186, 195]	[15, 36, 24, 35, 34, 31, 38, 47, 45, 46, 55, 77, 69, 56, 57, 59, 76, 79, 85, 94, 87, 63, 96, 61, 198, 190, 115, 120, 125, 83, 116, 201, 197, 157, 158, 163, 162, 178, 159, 183]
40	Z_9	[82, 10, 25, 29, 30, 31, 34, 19, 37, 41, 43, 45, 50, 40, 55, 57, 69, 71, 72, 73, 75, 78, 100, 93, 97, 89, 101, 117, 120, 122, 128, 127, 130, 132, 134, 137, 190, 165, 169, 143]	[0, 14, 15, 26, 31, 34, 36, 47, 44, 46, 54, 55, 76, 56, 57, 59, 77, 80, 87, 93, 85, 61, 94, 79, 111, 117, 120, 126, 128, 198, 183, 190, 173, 157, 177, 159, 188, 197, 203]

Table A2: Selected bands for each trial of McNemars test between SPPF and PSO-FCM on the Salinas Dataset.

n	Trial	SPPF	PSO-FCM
20	Z_0	[20, 34, 35, 36, 38, 40, 44, 47, 50, 52, 53, 59, 69, 80, 77, 89, 95, 97, 99, 100]	[5, 14, 18, 21, 24, 27, 29, 37, 47, 50, 53, 54, 61, 73, 82, 85, 88, 90, 97, 101]
20	Z_1	[20, 34, 35, 38, 40, 41, 43, 45, 47, 49, 51, 54, 67, 77, 81, 84, 88, 94, 98, 100]	[5, 11, 15, 22, 26, 29, 31, 37, 44, 47, 51, 53, 58, 61, 73, 85, 87, 89, 90, 96]
20	Z_2	[20, 36, 39, 42, 44, 46, 47, 49, 51, 53, 58, 65, 67, 72, 78, 87, 90, 92, 95, 99]	[5, 13, 15, 17, 24, 33, 38, 43, 47, 51, 53, 54, 61, 73, 86, 88, 89, 90, 94, 99]
20	Z_3	[14, 30, 32, 34, 38, 44, 47, 48, 50, 53, 58, 65, 70, 81, 77, 84, 87, 92, 96, 99]	[5, 13, 15, 21, 25, 28, 32, 38, 45, 47, 51, 72, 58, 71, 81, 82, 90, 94, 97, 101]
20	Z_4	[19, 33, 34, 36, 38, 40, 42, 45, 49, 52, 59, 66, 70, 75, 80, 87, 93, 95, 96, 99]	[14, 16, 18, 22, 25, 29, 35, 42, 44, 46, 52, 53, 60, 73, 85, 86, 89, 90, 94, 99]
20	Z_5	[12, 27, 28, 31, 35, 41, 45, 49, 52, 56, 61, 65, 67, 71, 78, 86, 88, 90, 95, 99]	[5, 14, 16, 20, 22, 24, 29, 37, 46, 50, 53, 54, 61, 73, 82, 87, 89, 90, 94, 98]
20	Z_6	[16, 30, 32, 33, 36, 39, 42, 45, 49, 51, 52, 56, 67, 75, 80, 82, 100, 85, 88, 95]	[5, 11, 14, 21, 26, 28, 29, 37, 45, 46, 51, 53, 62, 73, 82, 85, 87, 90, 94, 96]
20	Z_7	[12, 28, 31, 34, 36, 39, 42, 45, 50, 52, 54, 60, 70, 81, 77, 89, 95, 97, 98, 100]	[13, 15, 17, 22, 26, 29, 31, 36, 43, 46, 48, 52, 53, 61, 73, 82, 88, 90, 94, 98]
20	Z_8	[20, 35, 37, 41, 43, 44, 46, 50, 52, 55, 61, 66, 70, 76, 86, 92, 95, 97, 99, 101]	[5, 13, 16, 18, 22, 26, 28, 35, 44, 47, 50, 53, 60, 73, 83, 85, 88, 90, 94, 98]
20	Z_9	[17, 31, 32, 33, 37, 42, 45, 46, 47, 49, 54, 63, 67, 71, 80, 77, 89, 95, 99, 101]	[5, 10, 13, 16, 22, 28, 30, 36, 42, 46, 51, 52, 53, 60, 73, 85, 89, 90, 94, 98]
30	Z_0	[6, 14, 20, 23, 25, 27, 30, 34, 36, 38, 43, 46, 48, 50, 51, 54, 56, 59, 63, 66, 71, 80, 82, 77, 87, 91, 95, 98, 100, 101]	[12, 14, 15, 18, 21, 23, 25, 29, 31, 33, 34, 35, 40, 46, 47, 51, 53, 54, 58, 61, 64, 73, 82, 86, 89, 90, 91, 94, 96, 98]
30	Z_1	[10, 23, 25, 27, 28, 30, 34, 38, 43, 46, 48, 50, 51, 52, 54, 56, 61, 66, 69, 73, 76, 78, 81, 85, 88, 92, 95, 97, 99, 100]	[5, 11, 14, 16, 19, 22, 24, 26, 30, 33, 34, 35, 40, 47, 48, 50, 53, 60, 62, 68, 72, 73, 74, 77, 82, 85, 88, 89, 90, 96]
30	Z_2	[8, 20, 23, 25, 27, 29, 32, 35, 37, 39, 42, 46, 48, 50, 52, 53, 55, 61, 66, 68, 71, 76, 81, 84, 87, 90, 92, 95, 99, 101]	[5, 7, 10, 13, 16, 19, 21, 24, 25, 28, 30, 34, 39, 42, 45, 48, 51, 73, 53, 57, 60, 72, 82, 88, 89, 90, 94, 97, 99, 101]
30	Z_3	[10, 22, 24, 25, 27, 30, 34, 37, 39, 41, 44, 47, 49, 51, 52, 54, 56, 59, 64, 69, 75, 80, 86, 87, 89, 92, 95, 97, 99, 100]	[5, 10, 14, 18, 20, 22, 24, 28, 29, 31, 38, 44, 46, 48, 50, 52, 53, 62, 64, 72, 77, 78, 82, 83, 86, 87, 89, 90, 94, 98]
30	Z_4	[3, 13, 22, 24, 28, 30, 31, 32, 33, 35, 39, 44, 48, 50, 52, 54, 56, 61, 66, 68, 69, 73, 81, 78, 84, 87, 91, 95, 98, 100]	[5, 10, 13, 16, 22, 24, 25, 28, 30, 32, 34, 38, 44, 48, 51, 53, 54, 58, 62, 64, 73, 81, 82, 83, 87, 90, 91, 94, 98]
30	Z_5	[7, 17, 21, 23, 25, 26, 28, 32, 38, 42, 44, 47, 49, 51, 52, 53, 55, 57, 63, 70, 81, 77, 85, 87, 90, 92, 93, 95, 98, 100]	[4, 14, 16, 17, 18, 20, 24, 28, 30, 34, 36, 38, 41, 44, 46, 48, 51, 53, 54, 62, 64, 73, 82, 83, 88, 90, 92, 94, 98]
30	Z_6	[10, 22, 25, 27, 29, 30, 32, 35, 39, 41, 42, 44, 46, 47, 49, 54, 61, 65, 67, 70, 75, 81, 79, 85, 87, 92, 95, 97, 98, 100]	[5, 14, 16, 18, 21, 24, 28, 29, 32, 35, 38, 41, 46, 48, 50, 53, 54, 57, 60, 64, 73, 82, 86, 89, 90, 92, 94, 96, 97, 101]
30	Z_7	[11, 24, 26, 27, 29, 30, 32, 34, 37, 41, 45, 47, 49, 51, 52, 55, 58, 63, 66, 70, 74, 78, 81, 83, 85, 88, 90, 95, 99, 101]	[5, 14, 17, 18, 19, 21, 24, 26, 29, 33, 35, 37, 41, 44, 46, 48, 50, 53, 57, 58, 62, 72, 80, 82, 87, 89, 90, 94, 95, 97]
30	Z_8	[11, 26, 28, 29, 30, 31, 35, 40, 44, 47, 49, 50, 51, 52, 54, 55, 60, 66, 68, 70, 72, 75, 79, 82, 84, 89, 92, 94, 97, 100]	[0, 5, 7, 12, 15, 20, 24, 25, 28, 32, 34, 37, 38, 40, 44, 50, 51, 52, 53, 57, 60, 73, 83, 85, 87, 88, 89, 90, 94, 98]
30	Z_9	[10, 22, 24, 25, 28, 32, 35, 38, 39, 40, 43, 46, 49, 51, 53, 57, 64, 67, 69, 73, 77, 81, 83, 85, 87, 90, 94, 97, 99, 101]	[5, 12, 17, 18, 23, 26, 28, 29, 33, 37, 41, 42, 44, 46, 48, 50, 51, 53, 58, 62, 73, 86, 90, 91, 92, 94, 95, 97, 99, 101]
40	Z_0	[3, 7, 10, 15, 19, 22, 24, 29, 32, 34, 35, 36, 38, 42, 45, 47, 48, 50, 52, 53, 54, 56, 57, 63, 67, 69, 70, 74, 77, 79, 81, 83, 85, 86, 87, 91, 94, 96, 98, 100]	[0, 5, 8, 10, 12, 14, 17, 22, 25, 28, 29, 32, 34, 37, 39, 42, 43, 45, 48, 51, 52, 53, 54, 58, 61, 63, 70, 74, 75, 76, 78, 80, 82, 83, 85, 88, 89, 90, 91, 94]
40	Z_1	[1, 4, 7, 10, 15, 23, 26, 29, 34, 36, 38, 40, 42, 44, 46, 47, 49, 50, 52, 53, 55, 57, 61, 65, 67, 68, 72, 75, 79, 82, 84, 86, 88, 89, 90, 93, 95, 97, 98, 100]	[0, 5, 15, 16, 17, 20, 24, 25, 26, 29, 31, 33, 35, 36, 38, 41, 44, 48, 51, 53, 56, 58, 60, 62, 67, 69, 71, 72, 73, 74, 76, 77, 79, 82, 85, 89, 90, 94, 95, 99]
40	Z_2	[0, 1, 3, 12, 21, 23, 25, 27, 30, 32, 34, 35, 36, 38, 43, 49, 52, 54, 56, 58, 59, 60, 63, 66, 68, 71, 73, 75, 77, 79, 82, 84, 87, 90, 91, 93, 95, 98, 100, 101]	[0, 5, 10, 13, 15, 16, 18, 20, 22, 24, 28, 29, 32, 34, 35, 37, 40, 41, 42, 46, 50, 52, 53, 55, 58, 61, 63, 64, 71, 77, 79, 81, 82, 85, 86, 89, 90, 94, 96, 101]
40	Z_3	[6, 12, 16, 19, 21, 23, 26, 28, 30, 32, 34, 35, 37, 42, 44, 45, 46, 49, 52, 54, 56, 58, 63, 66, 68, 69, 72, 75, 77, 79, 82, 84, 85, 81, 88, 90, 94, 96, 97, 100]	[5, 9, 10, 11, 13, 16, 21, 25, 27, 30, 33, 34, 35, 40, 44, 46, 48, 53, 58, 61, 62, 65, 68, 69, 71, 73, 74, 75, 76, 77, 78, 79, 82, 88, 90, 91, 94, 95, 99]
40	Z_4	[0, 1, 2, 3, 11, 18, 22, 24, 26, 27, 29, 30, 33, 35, 37, 41, 44, 47, 48, 50, 54, 57, 60, 62, 64, 65, 68, 72, 74, 76, 77, 79, 82, 84, 89, 94, 96, 97, 99, 100]	[5, 6, 8, 10, 13, 18, 20, 22, 25, 28, 29, 31, 33, 35, 36, 39, 44, 48, 50, 51, 53, 55, 58, 60, 61, 69, 73, 74, 75, 76, 78, 79, 82, 87, 89, 90, 94, 95, 97, 101]
40	Z_5	[3, 6, 8, 14, 20, 24, 26, 29, 33, 35, 37, 39, 41, 44, 46, 47, 49, 51, 54, 58, 61, 64, 65, 67, 68, 69, 73, 75, 77, 79, 81, 83, 84, 89, 92, 94, 95, 97, 99, 100]	[5, 15, 17, 18, 19, 20, 23, 24, 25, 26, 29, 32, 34, 35, 38, 41, 44, 46, 47, 50, 53, 54, 57, 58, 61, 63, 69, 73, 74, 76, 80, 82, 85, 87, 90, 91, 94, 96, 99, 101]
40	Z_6	[5, 12, 15, 17, 19, 21, 23, 26, 29, 32, 34, 36, 38, 42, 45, 46, 47, 48, 51, 54, 58, 60, 62, 63, 65, 68, 71, 73, 75, 77, 79, 81, 84, 87, 89, 91, 95, 97, 99, 100]	[5, 7, 9, 13, 18, 20, 22, 23, 24, 26, 27, 31, 35, 36, 37, 39, 41, 42, 43, 45, 47, 50, 52, 53, 54, 56, 58, 60, 62, 72, 78, 80, 82, 83, 85, 89, 90, 94, 96, 99]
40	Z_7	[1, 4, 6, 12, 16, 19, 23, 25, 28, 30, 33, 36, 38, 39, 42, 46, 50, 54, 56, 57, 59, 62, 64, 66, 67, 68, 69, 72, 74, 76, 78, 81, 84, 87, 89, 90, 91, 95, 99, 101]	[0, 4, 5, 12, 15, 16, 18, 19, 20, 22, 24, 28, 31, 32, 35, 38, 41, 43, 47, 48, 51, 53, 57, 58, 60, 62, 63, 69, 73, 74, 76, 78, 80, 82, 87, 89, 90, 94, 97, 101]
40	Z_8	[3, 8, 10, 13, 17, 21, 23, 25, 29, 35, 37, 38, 39, 41, 43, 45, 50, 52, 54, 55, 56, 57, 58, 60, 64, 66, 69, 73, 75, 76, 79, 82, 83, 85, 86, 89, 94, 97, 99, 100]	[4, 5, 11, 14, 15, 18, 19, 20, 22, 24, 26, 28, 29, 31, 34, 37, 40, 43, 45, 46, 50, 53, 55, 57, 58, 61, 64, 69, 73, 74, 76, 78, 80, 82, 84, 86, 90, 91, 94, 98]
40	Z_9	[4, 8, 12, 18, 21, 23, 24, 25, 26, 28, 30, 33, 37, 41, 44, 47, 49, 51, 52, 53, 55, 59, 64, 66, 67, 69, 72, 74, 76, 80, 82, 78, 84, 87, 90, 92, 94, 96, 99, 101]	[5, 6, 11, 13, 14, 16, 19, 22, 24, 25, 29, 33, 35, 37, 40, 45, 48, 50, 51, 52, 53, 54, 58, 62, 68, 71, 73, 74, 75, 76, 78, 80, 82, 83, 86, 90, 91, 94, 96, 98]

Table A3: Selected bands for each trial of McNemars test between SPPF and PSO-FCM on the Pavia Center Dataset.

A2 Dataset Details

#	Class	Available Samples
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	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table A4: Available samples, Indian Pines

#	Class	Available Samples
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 A5: Available samples, Salinas

#	Class	Available Samples
1	Water	824
2	Trees	820
3	Asphalt	816
4	Self-Blocking Bricks	808
5	Bitumen	808
6	Tiles	1260
7	Shadows	476
8	Meadows	824
9	Bare Soil	820

Table A6: Available samples, Pavia Center