

REPUBLIC OF TURKEY  
YILDIZ TECHNICAL UNIVERSITY  
DEPARTMENT OF COMPUTER ENGINEERING



**CLASSIFICATION OF HYPERSPECTRAL IMAGES WITH  
DEEP METRIC LEARNING METHODS**

18011103 – Ömer Talha BAYSAN  
18011064 – Tolga SAĞLAM

**SENIOR PROJECT**

Advisor  
Prof. Dr. Gökhan BILGIN

January, 2024



## ACKNOWLEDGEMENTS

---

Dear Prof. Dr. Gökhan BILGIN,

First of all, we would like to express our sincere gratitude to you and Yıldız Technical University Computer Engineering department. With the support and education provided by you and your department, we have taken one of the most important steps of our lives.

We are grateful to you for the interest, patience and guidance you have shown us during this process. You have always given us a great vision by sharing current and advanced information with us students. We would also like to thank you for the importance and support you give to the development of students.

We would also like to express our gratitude to our families. The constant support and motivation they provide us is one of the greatest sources of power behind our success. We could not have completed this process successfully without the efforts and sacrifices of our families. That's why we offer our endless gratitude to our families.

Regards,

Ömer Talha BAYSAN  
Tolga SAĞLAM

## TABLE OF CONTENTS

---

<b>LIST OF ABBREVIATIONS</b>	<b>v</b>
<b>LIST OF FIGURES</b>	<b>vi</b>
<b>LIST OF TABLES</b>	<b>vii</b>
<b>ABSTRACT</b>	<b>viii</b>
<b>ÖZET</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Literature Review</b>	<b>3</b>
<b>3 Feasibility</b>	<b>8</b>
3.1 Technical Feasibility . . . . .	8
3.1.1 Software Feasibility . . . . .	8
3.1.2 Hardware Feasibility . . . . .	9
3.1.3 Communication Feasibility . . . . .	9
3.2 Workforce and Time Planning . . . . .	9
3.3 Economic Feasibility . . . . .	9
3.4 Legal Feasibility . . . . .	11
<b>4 System Analysis</b>	<b>12</b>
4.1 Measuring System Success . . . . .	12
<b>5 System Design</b>	<b>14</b>
5.1 Metric Learning . . . . .	17
5.1.1 Supervised Metric Learning . . . . .	18
5.1.2 Weakly Supervised Metric Learning . . . . .	19
5.1.3 Deep Metric Learning . . . . .	20
<b>6 Application</b>	<b>22</b>
<b>7 Experimental Results</b>	<b>25</b>

<b>8 Performance Analysis</b>	<b>28</b>
<b>9 Conclusion</b>	<b>31</b>
<b>References</b>	<b>32</b>
<b>Curriculum Vitae</b>	<b>34</b>

## LIST OF ABBREVIATIONS

---

HSI	Hyperspectral Imaging
ML	Machine Learning
CRF	Conditional Random Field
GIC	Grupo De Inteligencia Computacional
LMNN	Large Margin Nearest Neighbor Metric Learning
NCA	Neighborhood Components Analysis
KNN	k-nearest neighbors
LFDA	Local Fisher Discriminant Analysis
MLKR	Metric Learning for Kernel Regression
PCA	Principal Component Analysis
ITML	Information Theoretic Metric Learning
LSML	Logistic Discriminant Metric Learning
RCA	Relative Components Analysis
MMC	Mahalanobis Metric for Clustering

## LIST OF FIGURES

---

Figure 2.1	A hyperspectral image represented as a 3D cube with a point spectrum on the spectral cube represented in spatial position (x,y). . . . .	3
Figure 2.2	General scheme of HSI mapping of soil, vegetation and water. .	4
Figure 2.3	HSI-based fine classification of vegetable growing regions. . . .	5
Figure 4.1	Use Case Diagram . . . . .	12
Figure 5.1	Classes in the Pavia University Dataset . . . . .	14
Figure 5.2	BAND-60 of the Pavia University Dataset . . . . .	15
Figure 5.3	BAND-80 of the Pavia University Dataset . . . . .	15
Figure 5.4	BAND-100 of the Pavia University Dataset . . . . .	15
Figure 5.5	RGB Image of the Pavia University Dataset . . . . .	16
Figure 5.6	Nipy Sprectral Image of the Pavia University Dataset . . . . .	16
Figure 5.7	Nipy Sprectral Image of the Indian Pines Dataset . . . . .	17
Figure 6.1	NCA Model's Output for PAVIA University Dataset . . . . .	22
Figure 6.2	RCA Model's Output for PAVIA University Dataset . . . . .	22
Figure 6.3	Contrastive Loss Model's Output for PAVIA University Dataset .	23
Figure 6.4	NCA Model's Output for Indian Pines Dataset . . . . .	23
Figure 6.5	RCA Model's Output for Indian Pines Dataset . . . . .	24
Figure 6.6	Contrastive Loss Model's Output for Indian Pines Dataset . . . .	24
Figure 7.1	NCA Model Score Values with Pavia University Dataset . . . . .	25
Figure 7.2	LMNN Model Score Values with Pavia University Dataset . . . . .	26
Figure 7.3	ITML Model Score Values with Pavia University Dataset . . . . .	26
Figure 7.4	RCA Model Score Values with Pavia University Dataset . . . . .	26
Figure 7.5	NCA Model Score Values with Indian Pines Hyperspectral Dataset	27
Figure 7.6	LMNN Model Score Values with Indian Pines Hyperspectral Dataset . . . . .	27
Figure 7.7	ITML Model Score Values with Indian Pines Hyperspectral Dataset	27
Figure 7.8	RCA Model Score Values with Indian Pines Hyperspectral Dataset	27

## LIST OF TABLES

---

Table 3.1	Workforce - Timeline . . . . .	10
Table 3.2	Cost of the Team to Work on the Project . . . . .	10
Table 3.3	Cost of Hardware and Software to be Used in the Project . . . . .	10
Table 8.1	Models and Success Metric Scores . . . . .	28
Table 8.2	Supervised Learning Algorithms Models and Success Metric Scores	29
Table 8.3	Weakly Supervised Learning Algorithms Models and Success Metric Scores . . . . .	29
Table 8.4	Models and Success Metric Scores . . . . .	29
Table 8.5	Pavia University Dataset Models and Success Metric Scores . . . . .	30
Table 8.6	Indian Pines Hyperspectral Dataset Models and Success Metric Scores . . . . .	30



## ABSTRACT

---

# Classification of Hyperspectral Images with Deep Metric Learning Methods

Ömer Talha BAYSAN

Tolga SAĞLAM

Department of Computer Engineering

Senior Project

Advisor: Prof. Dr. Gökhan BILGIN

Classification of hyperspectral images with deep metric methods is one of the important topics in the fields of visual data analysis and artificial intelligence. These techniques have broad potential in remote sensing, medicine, environmental monitoring and many other applications. It is anticipated that effective classification of high-dimensional hyperspectral data will positively affect scientific and technological developments in these fields.

**Keywords:** Hyperspectral images, remote sensing, deep metric learning and classification

# Hiperspektral Görüntülerinin Derin Metrik Öğrenme Yöntemleri İle Sınıflandırılması

Ömer Talha BAYSAN

Tolga SAĞLAM

Bilgisayar Mühendisliği Bölümü

Bitirme Projesi

Danışman: Prof. Dr. Gökhan BILGIN

Hiperspektral görüntülerin derin metrik yöntemlerle sınıflandırılması görsel veri analizi ve yapay zeka alanlarının önemli konularından biridir. Bu tekniklerin uzaktan algılama, tıp, çevresel izleme ve diğer birçok uygulamada geniş bir potansiyeli vardır. Yüksek boyutlu hiperspektral verilerin etkili bir şekilde sınıflandırılması, bu alanlardaki bilimsel ve teknolojik gelişmeleri olumlu yönde etkilemesi ön görülmektedir.

**Anahtar Kelimeler:** Hiperspektral görüntüler, uzaktan algılama, derin metrik öğrenme ve sınıflandırma

# 1

## Introduction

---

Classification of hyperspectral images with deep metric methods is one of the important topics in the fields of visual data analysis and artificial intelligence. These techniques have broad potential in remote sensing, medicine, environmental monitoring and many other applications. Effective classification of high-dimensional hyperspectral data can positively impact scientific and technological developments in these fields.

The focus of this research is that despite the high resolution and detailed information provided by hyperspectral images, these data are inherently complex and high-dimensional [1]. The aim of this project is to maximize the potential of this technology by overcoming these challenges in classifying hyperspectral images. This effort reflects our motivation for how these techniques can help solve real-world problems.

Considering previous work in this field, significant progress has been made in the literature on the classification of hyperspectral images [2] [3]. However, this project aims to add a new dimension to the existing knowledge by focusing on how the use of deep metric methods addresses the deficiencies in this field.

The main research question of this study is how we can ensure accurate classification of hyperspectral images using deep metric methods. In this regard, we will focus on the following goals:

- To evaluate supervised learning and weakly supervised learning methods in classifying hyperspectral images.
- Improving classification performance using deep metric learning versions.

The methodological approach of the study will explain in detail how to apply deep metric methods to hyperspectral data sets and why these methods are preferred in this

study. In this context, a unique methodology will be created using programs such as OpenCV and Python, which include the basic principles in the field of image processing and machine learning.

The second chapter of the project report explains the definition of hyperspectral images and detailed information about them. It contains illustrative examples of the usage areas of hyperspectral images and their modern application areas. Previous projects on the classification of hyperspectral images are mentioned in detail.

In the third chapter of the report, the technical feasibility, business feasibility, economic feasibility and legal feasibility of the project were carried out.

The use case diagram of the project is shown in the fourth chapter. Success metrics used to measure system success are mentioned.

In the fifth chapter of the report, the details of the data set used in the project are explained and images from the data set are shared. The definition, content and algorithms used of metric learning, which is the focus of our project, were mentioned.

In the sixth chapter, the results of the models developed in the project are explained.

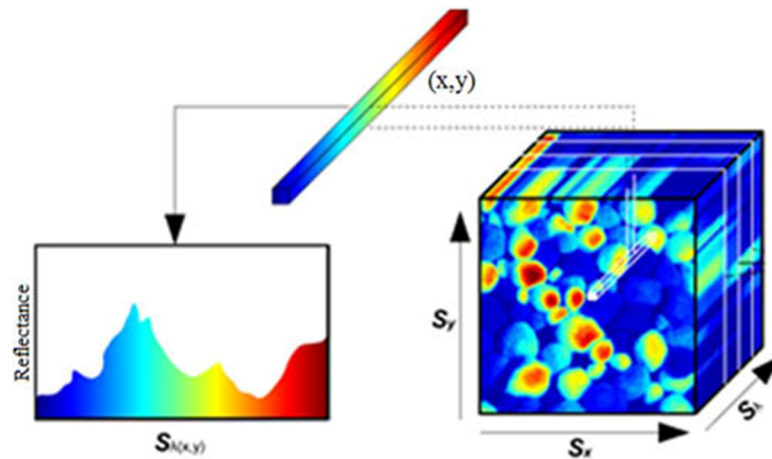
## 2 Literature Review

---

A literature review has been made in this chapter. It is explained what hyperspectral images are and in which areas and in what applications they are used. Additionally, articles in the project area are explained.

Hyperspectral images are remote sensing products that contain data measured in many narrow, contiguous wavelength bands covering a broad region of the electromagnetic spectrum. These images are used to examine and determine in detail the spectral properties of the surface materials of objects. Generally unlike multispectral images, hyperspectral images capture the spectral characteristics of each pixel at high resolution [4].

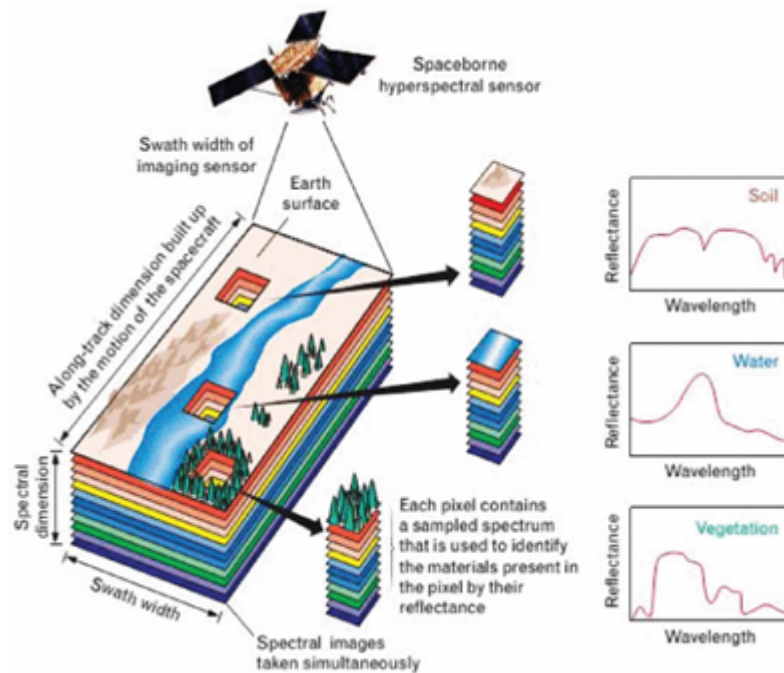
Figure 2.1 shows a hyperspectral image represented as a 3D cube with the point spectrum on the spectral cube represented in spatial location.



**Figure 2.1** A hyperspectral image represented as a 3D cube with a point spectrum on the spectral cube represented in spatial position  $(x,y)$ .

Substances on the Earth's surface absorb, transmit and reflect electromagnetic waves coming from the sun due to their properties. Thanks to hyperspectral sensors, the electromagnetic energy of substances is measured. As a result of deaths, the properties and changes of the substances on the earth's surface can be examined [5].

The reflectance values of substances on the Earth's surface can be plotted and compared in the electromagnetic spectrum range [6] . Figure 2.2 shows a general pattern of the spectral signatures of different materials found on the Earth's surface.



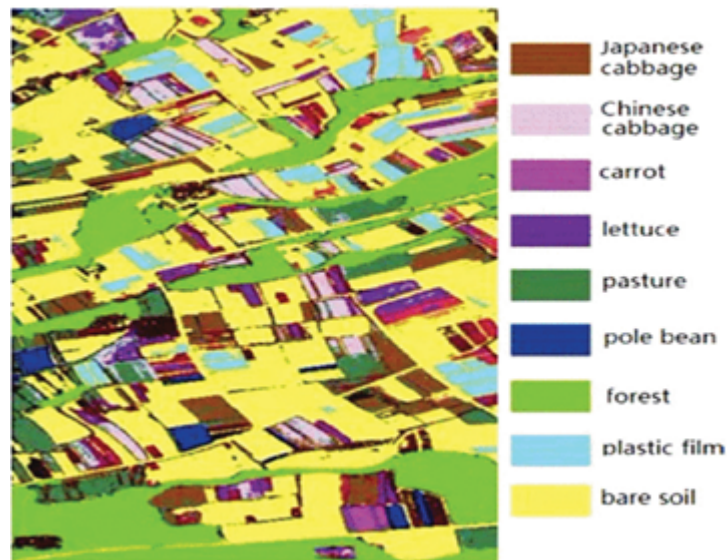
**Figure 2.2** General scheme of HSI mapping of soil, vegetation and water.

Hyperspectral imaging (HSI) is increasingly used in a wide variety of commercial, industrial and military applications. In this chapter, we focus on HSI's applications for food quality and safety, image-guided surgery and medical diagnosis, remote sensing such as precision agriculture and water resources management, forensic examination such as document forgery detection and artwork authentication, and defense and homeland security.

Hyperspectral images are used in a variety of fields, offering a wide range of applications. Here are some application areas of hyperspectral images:

**Agriculture and Plant Health Monitoring:** Hyperspectral images are used in agriculture to monitor plant health and increase agricultural productivity. By analyzing the spectral signatures of plants, it is possible to identify diseases, pests and nutritional status [7]. Figure 2.3 shows the HSI-based precise classification of vegetable growing regions.

**Environmental Monitoring:** In environmental applications, hyperspectral images are used to monitor various environmental parameters such as water quality, soil composition, forest cover, and the overall health of ecosystems.



**Figure 2.3** HSI-based fine classification of vegetable growing regions.

**Mineralogical Research:** In the mining industry and in the evaluation of natural resources, hyperspectral images are used to detect certain mineral types and map mineral reserves [8].

**Military Applications:** Hyperspectral images are used in military applications such as detecting camouflaged targets and identifying chemical and biological threats [9].

**Food Quality Control:** Hyperspectral images are used in the food industry to control product quality. Provides high-resolution spectral data to identify ingredients and quality of food products [10].

**Health and Medical Field:** In biomedical research, hyperspectral images are used to examine the spectral properties of tissue samples and biological materials. This is useful in medical diagnosis and research [11].

**Archaeology:** In analysis of historical sites and archaeological discoveries, hyperspectral images are used to detect subsoil structures and identify specific materials [12].

**Remote Sensing and Mapping:** In general, hyperspectral images are used in a wide range of remote sensing and mapping applications such as land use, natural resource management, disaster risk assessment, etc. [13].

The usage areas of hyperspectral images are diversifying and increasing day by day, thanks to the precise measurement of spectral differences between substances and situations.

Among the projects examined, one of the closest projects to the project to be realized is the project carried out in 2019 and whose subject is Hyperspectral Image Classification with Deep Metric Learning and Conditional Random Field [14]. This project proposes a new method for classifying objects in hyperspectral images, which are images that capture information beyond the visible spectrum.

Traditional methods for this task often require a lot of training data, which can be scarce and expensive to acquire. This project addresses this issue by combining two advanced techniques:

- Deep metric learning: This uses a neural network to learn features from the image's spectra that are tightly grouped within each object class.
- Conditional random field (CRF): This takes into account the spatial relationships between pixels in the image, helping to refine the classification based on context.

By combining these techniques, the project achieves good classification accuracy even with limited training data. The main steps involved are:

- Deep metric learning: The network extracts features from the image's spectra and maps them into a "feature space" where similar objects are close together.
- CRF inference: The CRF uses the learned features and the spatial relationships between pixels to refine the classification for each pixel.

This approach uses both spectral and spatial information from the image, leading to better accuracy than relying solely on spectral data. Additionally, the CRF helps to reduce noise and improve consistency in the classification results.

Overall, this project offers a promising solution for hyperspectral image classification, particularly when training data is limited.

The other project was carried out in 2015 and its subject is Spectral-Spatial Classification of Hyperspectral Data Based on Deep Belief Network [15]. This research project emphasizes the importance of hyperspectral data classification in remote sensing and acknowledges the recent attempts to tackle this challenge. However, conventional methods often extract surface-level features from the data, limiting their accuracy. This article proposes a novel deep learning approach for hyperspectral image classification using a deep belief network framework.

The key innovations include:



- Verification of suitability: The study confirms the effectiveness of restricted Boltzmann machines and deep belief networks for classification based on spectral information.
- Hybrid deep architecture: It introduces a new deep architecture that combines spectral-spatial feature extraction and classification for improved accuracy. This framework merges techniques like principal component analysis, hierarchical feature extraction, and logistic regression.
- Competitive performance: Experiments with real hyperspectral data demonstrate that the proposed classifier performs at par with the best existing methods.

Overall, the research highlights the immense potential of deep learning systems for achieving exceptional accuracy in hyperspectral data classification tasks.

# 3

## Feasibility

---

The high dimensionality of hyperspectral images often poses a challenge to traditional data analysis techniques. In order to improve the classification performance of hyperspectral images, it is often introduced in metric learning to assign small distances between samples from the same class and large distances from different class. However, most traditional metric learning methods only adopt linear transformations, which cannot capture complex nonlinear relationships between high-dimensional samples. In this project, it is aimed to use metric learning methods by using spectral and spatial information to increase the reliability of the classification results for the classification of hyperspectral images.

### 3.1 Technical Feasibility

Technical Feasibility includes Software, Hardware and Communication Feasibility.

#### 3.1.1 Software Feasibility

The programming language used in the project is Python, a common language used in the fields of machine learning and artificial intelligence. Python language was chosen for the project because it is simple, understandable and versatile. Python's unique features are extensive library support, easy readability and ease of learning, community support and documentation, speed and performance, and the ability to run smoothly on different operating systems. The project-specific benefits provide a wide set of tools for data processing, deep learning, visualization and analysis. The documentation provided for metric learning and deep metric learning libraries makes the Python programming language perfect for our project.

Sklearn and Metric-learning libraries were used for machine learning and metric learning in the project. While Sklearn includes a large collection of machine learning tools, its metric learning library allows the development of algorithms focused on

similarity measurement. These libraries are at the heart of the project, supporting different learning and measurement techniques and allowing developers to work with a wide range of tools.

Models will be trained using Google Collaboratory. It is a cloud-based machine learning platform that is free for developers. It is one of the ideal choices for the project thanks to its easy-to-understand interface and powerful GPU resources.

### **3.1.2 Hardware Feasibility**

This project covers the fields of natural language processing, image processing and machine learning. Since these areas require high performance, computers with high hardware features are needed. Google Collaboratory, which has free high RAM and powerful graphics cards, will be used in the project.

Google Collaboratory works with Intel Xeon CPU @2.20 GHz processor, 13 GB RAM, Tesla K80 graphics card and 12 GB GDDR5 VRAM system features. The computer owned by the project team works with Intel i5 - 1135G7 processor, 16 GB RAM, 2GB Nvidia MX 2060 graphics card system features. While the data set is trained on Google Collaboratory, it will also be stored in Google Drive, a cloud-based data storage platform, and in the storage space on our personal computer in case of adverse events.

### **3.1.3 Communication Feasibility**

The project team will hold regular online meetings via Zoom and Discord platforms, as only the graduation project course remains during the training periods. In these meetings, progress on the project, sharing of ideas and any problems that need solutions will be discussed. With this approach, team members will stay in touch and carry out the project efficiently.

## **3.2 Workforce and Time Planning**

It is anticipated that the project will last 3 months. Team members will carry out the project development process in a modular manner, as shown in Figure 3.1.

## **3.3 Economic Feasibility**

Since our project will use the Google Collaboratory platform, which is free to access, it will reduce the cost required for hardware. Tensorflow software is open source and can be used free of charge. In addition, since the University of PAVIA's Hyperspectral

**Table 3.1** Workforce - Timeline

İP No	Work Palette Name	WEEKS (02.10.2023 – 15.01.2024)															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	Meeting with the Advisor	X															
2	Project Research		X	X	X	X											
3	Dataset Review					X	X										
4	Preparation of Interim Report						X	X									
5	Creating the Basic Model						X	X	X								
6	Model Optimization									X	X						
7	Creating Different Models										X	X					
8	Optimization of Different Models												X	X			
9	Final Report													X	X	X	X

Imagery dataset, which is offered free of charge to developers, will be used, there will be no need to spend additional resources to collect data.

**Table 3.2** Cost of the Team to Work on the Project

Name	Weekly Working Time	Project Duration	Monthly Fee	Overall Cost
Ömer Talha BAYSAN	18 Hours	3 Months	9000 TL	27000 TL
Tolga SAĞLAM	18 Hours	3 Months	9000 TL	27000 TL

Considering the costs in table 3.2, the amount to be paid to the project team is 54000 TL.

**Table 3.3** Cost of Hardware and Software to be Used in the Project

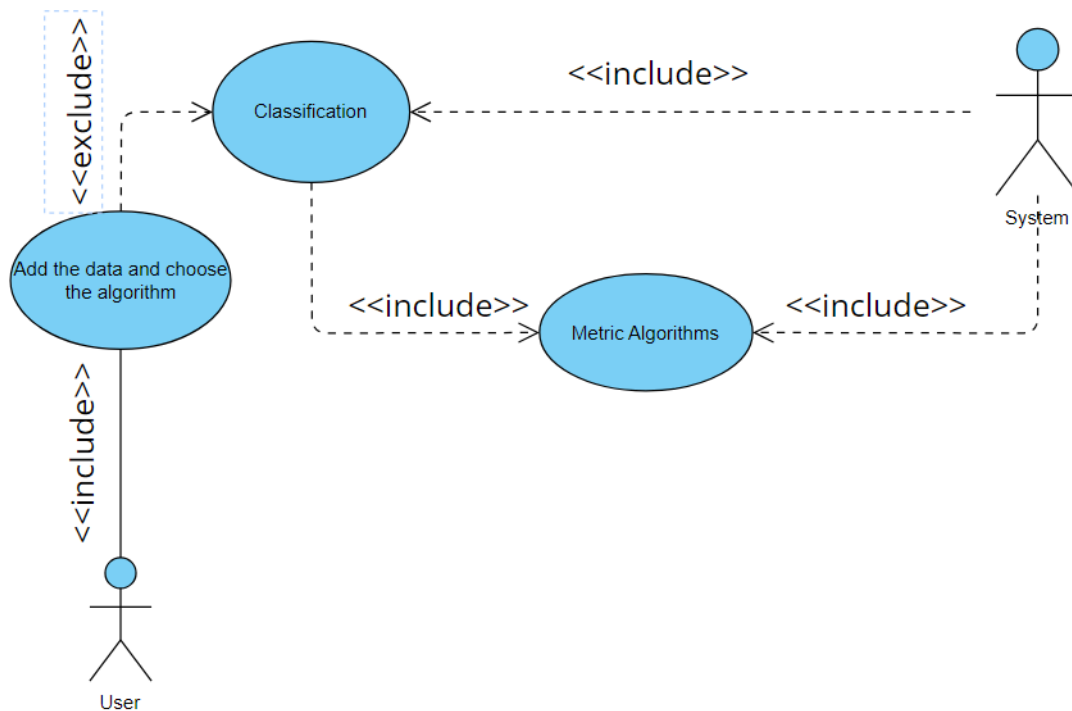
Product	Price
Computer Equipped with Intel i5 - 1135G7 processor, 512GB SSD 16 GB RAM 2 GB Nvidia MX450 GPU	18500 TL
Microsoft Windows 10	3500 TL

The cost of the hardware and software to be used in the project is shown in Table 3.3. Considering the costs in the table, the amount to be paid for hardware and software requirements is 22000 TL. When the total project cost is calculated, the cost of the project is 76000 TL. When the costs are examined, it is predicted that the project can generate income if licensed.

### **3.4 Legal Feasibility**

The University of PAVIA's Hyperspectral Imagery dataset is freely available for research purposes. Open source software 'sklearn' and 'metric-learning' are available free of charge. Google Collaboratory is a cloud-based platform offered by Google. The software used in the project is used in accordance with the law for the project. In addition, data protection laws have been taken into account and the project is expected to be completed successfully without encountering any legal problems.

The functioning of the system begins with the user inputting data. Subsequently, one of the metric algorithms is chosen, and a model is selected to carry out the classification.



**Figure 4.1** Use Case Diagram

### 4.1 Measuring System Success

In our model, accuracy, which stands out as one of the most well-known validation methods in Machine Learning models, has been utilized for the evaluation of classification problems. Its popularity can be attributed to its relative simplicity, being easy to understand and implement. In straightforward scenarios, accuracy serves as an effective metric for evaluating model performance.

Yet, real-life modeling scenarios are seldom straightforward. Dealing with imbalanced datasets or engaging in multiclass or multilabel classification problems may become necessary. There are instances where achieving high accuracy may not align with your objectives. As you tackle more intricate ML challenges, the calculation and utilization of accuracy become less straightforward and demand additional consideration.

In the realm of classification problems, accuracy serves as a metric to quantify the percentage of accurate predictions generated by a model. The accuracy score in machine learning gauges the model's performance by assessing the ratio of correct predictions to the total number of predictions. This score is calculated by dividing the number of accurate predictions by the overall number of predictions.

$$Accuracy = \frac{NumberofCorrectPredictions}{TotalNumberofPredictions} \quad (4.1)$$

## 5 System Design

---

In the project, two different data sets have been utilized. These data sets are the Indian Pines and Pavia University data sets. PAVIA University's dataset was acquired during a flight campaign over PAVIA using the Rosis sensor. The images were made publicly available by Grupo De Inteligencia Computacional (GIC). The hyperspectral imaging (HSI) data comprises high-resolution spectral information at 103 different wavelengths and has a resolution of 610\*340 pixels. The data is divided into 10 distinct classes, labeled from 0 to 9. Pavia University indicated that class 0 represents empty or void data, so these instances were excluded during preprocessing before analysis. The geometric resolution is 1.3 meters. The distribution of the 9 remaining classes in the dataset is as follows:

Label	Class	Samples
1	Asphalt	6631
2	Meadows	18649
3	Gravel	2099
4	Trees	3064
5	Painted metal sheets	1345
6	Bare Soil	5029
7	Bitumen	1330
8	Self-Blocking Bricks	3682
9	Shadows	947

**Figure 5.1** Classes in the Pavia University Dataset

Below are some sample band images of the dataset:





**Figure 5.2** BAND-60 of the Pavia University Dataset



**Figure 5.3** BAND-80 of the Pavia University Dataset



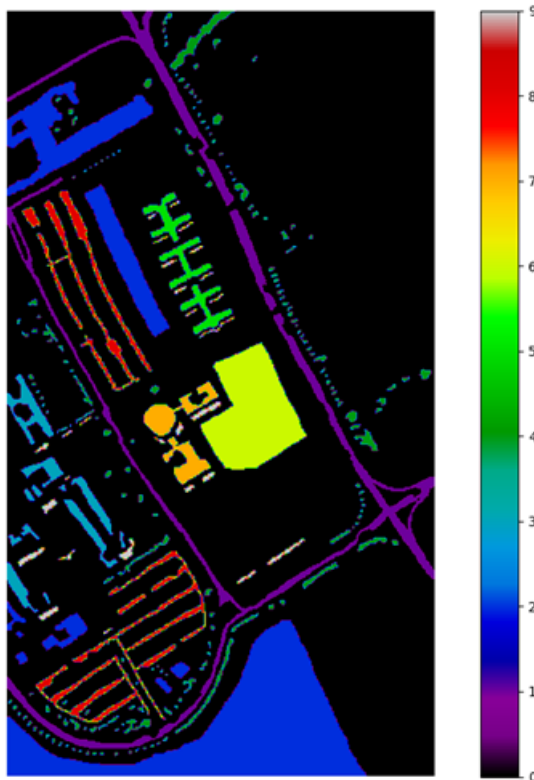
**Figure 5.4** BAND-100 of the Pavia University Dataset

Subsequently, the image of the data in the (60, 27, 40) RGB color scale has been extracted.



**Figure 5.5** RGB Image of the Pavia University Dataset

The image, segmented into classes using the `nipy_spectral` color map for the data, is as follows:



**Figure 5.6** Nipy Spectral Image of the Pavia University Dataset

Indian Pines dataset's resolution is  $145 \times 145$  pixels. The data is divided into 16 distinct classes. The distribution of the 16 classes in the dataset is as follows:



**Figure 5.7** Nipy Spectral Image of the Indian Pines Dataset

## 5.1 Metric Learning

In machine learning, certain approaches are necessary to measure the distance between data points. Traditionally used distance metrics include metrics like Euclidean, Manhattan, and Cosine. Metric learning aims to automatically generate task-specific distance metrics, representing a machine learning approach often relying on supervised data. The fundamental goal of metric learning is to bring similar examples closer together while pushing different examples farther apart. This learned distance metric can be employed in various tasks such as K-NN classification, clustering, information retrieval, and more, to achieve effective results. Real-world applications where this learned metric can be utilized include facial recognition, object recognition, scaling, embedding learning, and so on [16].

Metric learning problems are broadly categorized into two main types based on the available type of supervision in the training data: Supervised and Weakly supervised.

### **5.1.1 Supervised Metric Learning**

The data points that the algorithm has access to consist of a set of data, each belonging to a specific class (label); this situation is similar to a typical classification problem. Generally, the objective in this context is to bring points with the same label close to each other and place points with different labels at a distant distance.

#### **5.1.1.1 LMNN (Large Margin Nearest Neighbor Metric Learning)**

LMNN is a supervised metric learning algorithm used to enhance the accuracy of the k-Nearest Neighbors algorithm by learning the Mahalanobis distance. This algorithm is based on a subfield of convex optimization within the scope of semi-definite programming. The LMNN method aims to learn a metric with the goal of making k-nearest neighbors belong to the same class as much as possible and separating examples from different classes with a wide margin. Mahalanobis distances, calculated with a linear transformation of the input space, are computed using Euclidean distances in the transformed space. Euclidean distances in the transformed space can be considered as Mahalanobis distances [16].

The linear transformation of the input space is achieved through an optimization process involving two terms designed to minimize a loss function. The first term strengthens the relationship between matching class examples by reducing distances between k-nearest neighbors. On the other hand, the second term focuses on making distances between examples from different classes larger, thereby separating classes more distinctly. Regulating these terms performs a linear transformation of the input space, increasing the number of k-nearest neighbors of training examples that have matching classes [16].

#### **5.1.1.2 NCA (Neighborhood Components Analysis)**

NCA is a distance metric learning algorithm used to enhance the accuracy of k-Nearest Neighbors classification. It aims to reach the target more effectively compared to the standard Euclidean distance. The algorithm adopts a direct stochastic variant that sequentially removes each training example (leave-one-out) to maximize the score of k-nearest neighbors (KNN). Additionally, due to its ability to learn a low-dimensional linear transformation of the data, the algorithm can be employed for data visualization and fast classification purposes [16].

#### **5.1.1.3 LFDA (Local Fisher Discriminant Analysis)**

LFDA stands out as a linear supervised dimensionality reduction method that successfully combines the concepts of Linear Discriminant Analysis and Local Preserving Projection, offering practical utility, especially when dealing with multiple modalities. In this scenario, one or more classes are found as discrete clusters in the input space. The fundamental optimization problem of LFDA is solved by treating it as a generalized eigenvalue problem [16].

#### **5.1.1.4 MLKR (Metric Learning for Kernel Regression)**

MLKR is one of the supervised metric learning algorithms that directly minimizes the regression error to learn the distance function. This method can be seen as a supervised derivative of PCA. However, it can also be utilized for dimensionality reduction when visualizing high-dimensional data [16].

### **5.1.2 Weakly Supervised Metric Learning**

The algorithm has access only to supervision information at the tuple level (usually in the form of pairs, triplets, or quadruplets). A classic example of weak supervision includes a set of positive and negative pairs. The objective here is to learn a distance metric that keeps positive pairs close to each other and negative pairs at a distant distance [16].

#### **5.1.2.1 ITML (Information Theoretic Metric Learning)**

ITML minimizes the (differential) relative entropy, i.e., the Kullback–Leibler divergence, between two multivariate Gaussian distributions, taking constraints on the Mahalanobis distance into account. Transforming the LogDet divergence into linear constraints enables it to be converted into a Bregman optimization problem, allowing the divergence to be minimized. This algorithm can handle various types of constraints and can optionally incorporate a prior state into the distance function. Notably, ITML distinguishes itself from some other methods by not relying on eigendecomposition or semi-definite programming [16].

#### **5.1.2.2 LSML (Logistic Discriminant Metric Learning)**

LSML selects two instances from the same class and two instances from different classes to create labeled instances. In this way, the first two points in the resulting quads are intended to be more similar than the last two points [16].

#### **5.1.2.3 RCA (Relative Components Analysis)**

RCA (Relevance Component Analysis) learns a fully ranked Mahalanobis distance metric based on the weighted sum of in-chunklets covariance matrices. It applies a general linear transformation with the goal of assigning high weights to relevant dimensions while assigning low weights to irrelevant dimensions. The relevant dimensions are predicted using subsets called 'chunklets,' where known points belong to the same class [16].

#### **5.1.2.4 MMC (Mahalanobis Metric for Clustering )**

MMC focuses on minimizing the sum of squared distances between similar points while constraining the sum of distances between dissimilar points to be greater than one. This leads to a convex optimization problem that can be effectively solved, resulting in a structure free from local minima. However, the algorithm fundamentally involves eigendecomposition, which becomes a primary speed-limiting factor. Originally developed for clustering applications, MMC carries an implicit assumption: it assumes that all classes tightly cluster together, following a single mode, and therefore have a unimodal distribution. This assumption limits the applicability of the method. Nevertheless, MMC remains a widely referenced and pioneering technique [16].

### **5.1.3 Deep Metric Learning**

In traditional machine learning methodologies, raw data has limitations in processing capacity. Therefore, feature engineering is required before proceeding to classification or clustering, i.e. preprocessing and feature extraction. These processes often require expertise and are not automatically covered by classification systems. Deep learning, on the other hand, has the ability to learn high-level data directly within the data categorization structure. The main difference between classical machine learning approaches and deep learning can be clearly seen in this perspective [17].

#### **5.1.3.1 Contrastive Loss**

Contrastive Loss is a technique that learns common features and enhances features that distinguish one class of data from another, using a principle based on comparison between classes of data, in order to improve performance in vision tasks. Contrastive Loss is a machine learning paradigm that aims to learn from unlabeled data points how a model should distinguish between similar and dissimilar points. In contrastive loss, samples are compared and those belonging to the same distribution are approximated in the embedding space. On the other hand, those belonging to different distributions

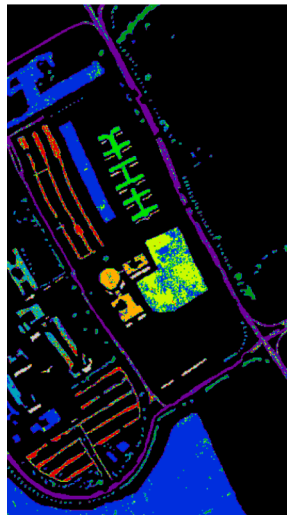
are pulled against each other [17].

# 6

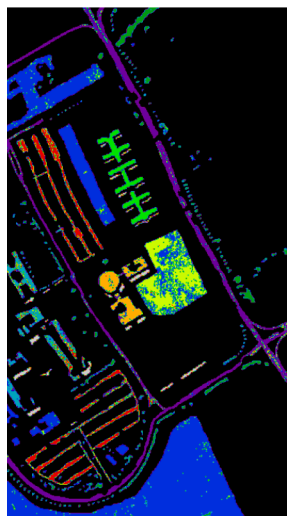
## Application

---

The outputs of the NCA, RCA, and Contrastive loss metrics for the PAVIA university dataset can be observed in the sample visuals in Figure 6.1, Figure 6.2, and Figure 6.3, respectively.



**Figure 6.1** NCA Model's Output for PAVIA University Dataset



**Figure 6.2** RCA Model's Output for PAVIA University Dataset

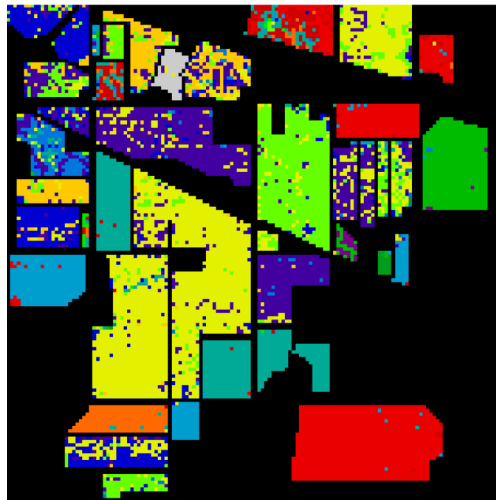




**Figure 6.3** Contrastive Loss Model's Output for PAVIA University Dataset

When examining the visuals, it can be observed that Contrastive loss demonstrates a highly successful performance compared to NCA and RCA. Upon closer inspection of NCA and RCA, it can be seen that NCA performs slightly better than RCA.

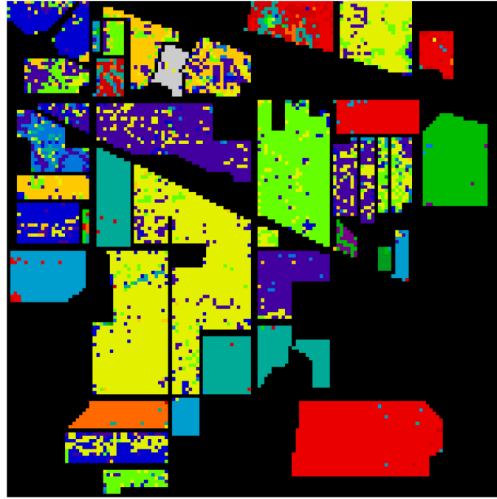
The outputs of the NCA, RCA, and Contrastive loss metrics for the Indian Pines dataset can be observed in Figure 6.4, Figure 6.5, and Figure 6.6, respectively.



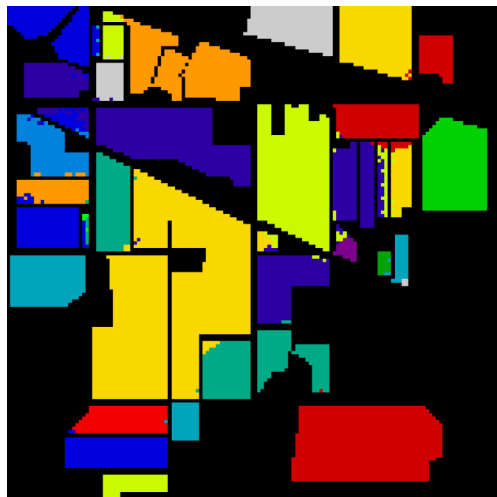
**Figure 6.4** NCA Model's Output for Indian Pines Dataset

Upon examining these three visuals, it is evident that Contrastive loss consistently exhibits excellent performance. NCA and RCA, on the other hand, yield very similar results to each other.

It can be easily observed that the metrics perform better on the PAVIA University dataset than on the Indian Pines dataset. This is mainly because the PAVIA University dataset has much more data than the Indian Pines dataset.



**Figure 6.5** RCA Model's Output for Indian Pines Dataset



**Figure 6.6** Contrastive Loss Model's Output for Indian Pines Dataset

## 7 Experimental Results

---

In the project, the effectiveness of metric learning methods, supervised learning algorithms and weakly supervised learning algorithms in the classification of hyperspectral images was investigated. In the project, models were developed by applying LMNN, NCA, LFDA and MLKR supervised learning algorithms and ITML, LSML, MMC, RCA weakly supervised learning algorithms on the University of Pavia hyperspectral image dataset.

During the training phase, we processed the dataset with the K-fold cross-validation method. K-fold is a validation technique used to evaluate the performance of the model by splitting the data set into training and testing subsets. In this method, the data set was divided into five equal parts and each part was used as the test set and the remaining four parts were used as the train set.

According to the experimental results obtained, the results regarding the performance of LMNN and NCA algorithms are visually presented in Figure 7.1 and Figure 7.2.

```
Fold 1 -> Train: 8556, Test: 34220
NCA accuracy on test set of 34220 points: 0.8830
Fold 2 -> Train: 8555, Test: 34221
NCA accuracy on test set of 34221 points: 0.8795
Fold 3 -> Train: 8555, Test: 34221
NCA accuracy on test set of 34221 points: 0.8840
Fold 4 -> Train: 8555, Test: 34221
NCA accuracy on test set of 34221 points: 0.8865
Fold 5 -> Train: 8555, Test: 34221
NCA accuracy on test set of 34221 points: 0.8845
Veri kümesinin ortalaması: 0.8835152860619198
Veri kümesinin standart sapması: 0.0022964241051127918
```

**Figure 7.1** NCA Model Score Values with Pavia University Dataset

According to the experimental results obtained, the results regarding the performance of ITML and RCA algorithms are presented visually in Figure 7.3 and Figure 7.4.

In the project, the effectiveness of metric learning methods, supervised learning algorithms and weakly supervised learning algorithms in the classification of

```

Fold 1 -> Train: 8556, Test: 34220
LMNN accuracy on test set of 34220 points: 0.9185
Fold 2 -> Train: 8555, Test: 34221
LMNN accuracy on test set of 34221 points: 0.9182
Fold 3 -> Train: 8555, Test: 34221
LMNN accuracy on test set of 34221 points: 0.9209
Fold 4 -> Train: 8555, Test: 34221
LMNN accuracy on test set of 34221 points: 0.9192
Fold 5 -> Train: 8555, Test: 34221
LMNN accuracy on test set of 34221 points: 0.9254
Veri kümesinin ortalaması: 0.9204460331255919
Veri kümesinin standart sapması: 0.00265909151389906

```

**Figure 7.2** LMNN Model Score Values with Pavia University Dataset

```

Fold 1 -> Train: 8556, Test: 34220
ITML_Supervised accuracy on test set of 34220 points: 0.6563
Fold 2 -> Train: 8555, Test: 34221
ITML_Supervised accuracy on test set of 34221 points: 0.4661
Fold 3 -> Train: 8555, Test: 34221
ITML_Supervised accuracy on test set of 34221 points: 0.5243
Fold 4 -> Train: 8555, Test: 34221
ITML_Supervised accuracy on test set of 34221 points: 0.6810
Fold 5 -> Train: 8555, Test: 34221
ITML_Supervised accuracy on test set of 34221 points: 0.5664
Veri kümesinin ortalaması: 0.5788238581786204
Veri kümesinin standart sapması: 0.08032347646446751

```

**Figure 7.3** ITML Model Score Values with Pavia University Dataset

```

Fold 1 -> Train: 8556, Test: 34220
RCA_Supervised accuracy on test set of 34220 points: 0.8830
Fold 2 -> Train: 8555, Test: 34221
RCA_Supervised accuracy on test set of 34221 points: 0.8795
Fold 3 -> Train: 8555, Test: 34221
RCA_Supervised accuracy on test set of 34221 points: 0.8840
Fold 4 -> Train: 8555, Test: 34221
RCA_Supervised accuracy on test set of 34221 points: 0.8781
Fold 5 -> Train: 8555, Test: 34221
RCA_Supervised accuracy on test set of 34221 points: 0.8819
Veri kümesinin ortalaması: 0.8813178049830501
Veri kümesinin standart sapması: 0.0021985823674317314

```

**Figure 7.4** RCA Model Score Values with Pavia University Dataset

hyperspectral images was investigated. In the project, models were developed by applying LMNN, NCA, LFDA and MLKR supervised learning algorithms and ITML, LSML, MMC, RCA weakly supervised learning algorithms on the Indian Pines Hyperspectral Dataset.

According to the experimental results obtained, the results regarding the performance of LMNN and NCA algorithms are visually presented in Figure 7.5 and Figure 7.6.

According to the experimental results obtained, the results regarding the performance of ITML and RCA algorithms are visually presented in Figure 7.7 and Figure 7.8.

In this project, we evaluate the potential of deep metric learning methods in

```

Fold 1 -> Train: 8199, Test: 2050
NCA accuracy on test set of 2050 points: 0.7517
Fold 2 -> Train: 8199, Test: 2050
NCA accuracy on test set of 2050 points: 0.7702
Fold 3 -> Train: 8199, Test: 2050
NCA accuracy on test set of 2050 points: 0.7659
Fold 4 -> Train: 8199, Test: 2050
NCA accuracy on test set of 2050 points: 0.7639
Fold 5 -> Train: 8200, Test: 2049
NCA accuracy on test set of 2049 points: 0.7721
Veri kümesinin ortalaması: 0.7647582520920377
Veri kümesinin standart sapması: 0.007155659981174996

```

**Figure 7.5** NCA Model Score Values with Indian Pines Hyperspectral Dataset

```

Fold 1 -> Train: 8199, Test: 2050
LMNN accuracy on test set of 2050 points: 0.8727
Fold 2 -> Train: 8199, Test: 2050
LMNN accuracy on test set of 2050 points: 0.8717
Fold 3 -> Train: 8199, Test: 2050
LMNN accuracy on test set of 2050 points: 0.8785
Fold 4 -> Train: 8199, Test: 2050
LMNN accuracy on test set of 2050 points: 0.8722
Fold 5 -> Train: 8200, Test: 2049
LMNN accuracy on test set of 2049 points: 0.8755
Veri kümesinin ortalaması: 0.8741341999071528
Veri kümesinin standart sapması: 0.0025744424990030916

```

**Figure 7.6** LMNN Model Score Values with Indian Pines Hyperspectral Dataset

```

Fold 1 -> Train: 8199, Test: 2050
ITML_Supervised accuracy on test set of 2050 points: 0.6288
Fold 2 -> Train: 8199, Test: 2050
ITML_Supervised accuracy on test set of 2050 points: 0.6102
Fold 3 -> Train: 8199, Test: 2050
ITML_Supervised accuracy on test set of 2050 points: 0.5478
Fold 4 -> Train: 8199, Test: 2050
ITML_Supervised accuracy on test set of 2050 points: 0.5951
Fold 5 -> Train: 8200, Test: 2049
ITML_Supervised accuracy on test set of 2049 points: 0.6286
Veri kümesinin ortalaması: 0.6021101072504136
Veri kümesinin standart sapması: 0.029923363065741195

```

**Figure 7.7** ITML Model Score Values with Indian Pines Hyperspectral Dataset

```

Fold 1 -> Train: 8199, Test: 2050
RCA_Supervised accuracy on test set of 2050 points: 0.7517
Fold 2 -> Train: 8199, Test: 2050
RCA_Supervised accuracy on test set of 2050 points: 0.7702
Fold 3 -> Train: 8199, Test: 2050
RCA_Supervised accuracy on test set of 2050 points: 0.7659
Fold 4 -> Train: 8199, Test: 2050
RCA_Supervised accuracy on test set of 2050 points: 0.7639
Fold 5 -> Train: 8200, Test: 2049
RCA_Supervised accuracy on test set of 2049 points: 0.7735
Veri kümesinin ortalaması: 0.7650510778607054
Veri kümesinin standart sapması: 0.007472399872753166

```

**Figure 7.8** RCA Model Score Values with Indian Pines Hyperspectral Dataset

hyperspectral image classification and shed light on future work in this field by presenting a comparative analysis of different algorithms.

## 8 Performance Analysis

Table 8.6 shows the success metrics of the 18 models prepared.

**Table 8.1** Models and Success Metric Scores

Models	Accuracy Score	F1 Score	Kappa Score	Mean and Standard Deviation
LFDA-P	0.892	0.872	0.854	0.894 +- 0.004
LFDA-I	0.803	0.791	0.771	0.810 +- 0.009
LMNN-P	0.925	0.879	0.899	0.920 +- 0.002
LMNN-I	0.875	0.835	0.856	0.874 +- 0.002
MLKR-P	0.884	0.867	0.843	0.881 +- 0.002
MLKR-I	0.773	0.703	0.739	0.774 +- 0.007
NCA-P	0.884	0.867	0.843	0.883 +- 0.002
NCA-I	0.772	0.702	0.737	0.764 +- 0.007
ITML-P	0.634	0.452	0.497	0.666 +- 0.079
ITML-I	0.628	0.545	0.568	0.602 +- 0.029
LSML-P	0.906	0.895	0.875	0.907 +- 0.001
LSML-I	0.773	0.703	0.739	0.765 +- 0.007
MMC-P	0.855	0.834	0.803	0.851 +- 0.008
MMC-I	0.754	0.664	0.716	0.743 +- 0.020
RCA-P	0.881	0.865	0.839	0.881 +- 0.002
RCA-I	0.783	0.714	0.749	0.775 +- 0.007
DEEP-P	0.996	0.994	0.994	0.991 +- 0.324
DEEP-I	0.987	0.974	0.984	0.977 +- 1.470

In Table 8.6, the expression "-P" next to the algorithm names means that the Pavia University Dataset was used when developing the models, while the expression "-I" means that the Indian Pines Hyperspectral Dataset was used.

When all the scores are evaluated, it is clearly seen that the deep metric learning method is the most successful method.

Table 8.2 shows the success scores of Supervised Learning Algorithms models. When looking at both datasets, it can be seen that the LMNN algorithm is the most successful

**Table 8.2** Supervised Learning Algorithms Models and Success Metric Scores

Models	Accuracy Score	F1 Score	Kappa Score	Mean and Standard Deviation
LFDA-P	0.892	0.872	0.854	0.894 +- 0.004
LFDA-I	0.803	0.791	0.771	0.810 +- 0.009
LMNN-P	0.925	0.879	0.899	0.920 +- 0.002
LMNN-I	0.875	0.835	0.856	0.874 +- 0.002
MLKR-P	0.884	0.867	0.843	0.881 +- 0.002
MLKR-I	0.773	0.703	0.739	0.774 +- 0.007
NCA-P	0.884	0.867	0.843	0.883 +- 0.002
NCA-I	0.772	0.702	0.737	0.764 +- 0.007

algorithm compared to other algorithms.

**Table 8.3** Weakly Supervised Learning Algorithms Models and Success Metric Scores

Models	Accuracy Score	F1 Score	Kappa Score	Mean and Standard Deviation
ITML-P	0.634	0.452	0.497	0.666 +- 0.079
ITML-I	0.628	0.545	0.568	0.602 +- 0.029
LSML-P	0.906	0.895	0.875	0.907 +- 0.001
LSML-I	0.773	0.703	0.739	0.765 +- 0.007
MMC-P	0.855	0.834	0.803	0.851 +- 0.008
MMC-I	0.754	0.664	0.716	0.743 +- 0.020
RCA-P	0.881	0.865	0.839	0.881 +- 0.002
RCA-I	0.783	0.714	0.749	0.775 +- 0.007

Table 8.3 shows the success scores of Weakly Supervised Learning Algorithms models. When looking at both datasets, it can be seen that the LSML algorithm is the most successful algorithm compared to other algorithms.

**Table 8.4** Models and Success Metric Scores

Models	Accuracy Score	F1 Score	Kappa Score	Mean and Standard Deviation
DEEP-P	0.996	0.994	0.994	0.991 +- 0.324
DEEP-I	0.987	0.974	0.984	0.977 +- 1.470

Table 8.4 shows the success results of deep metric learning approaches. When the success of the Pavia University Dataset and the Indian Pines Hyperspectral Dataset is compared, it is seen that the Pavia University Dataset is more successful in the models.

**Table 8.5** Pavia University Dataset Models and Success Metric Scores

Models	Accuracy Score	F1 Score	Kappa Score	Mean and Standard Deviation
LFDA	0.892	0.872	0.854	0.894 +- 0.004
LMNN	0.925	0.879	0.899	0.920 +- 0.002
MLKR	0.884	0.867	0.843	0.881 +- 0.002
NCA	0.884	0.867	0.843	0.883 +- 0.002
ITML	0.634	0.452	0.497	0.666 +- 0.079
LSML	0.906	0.895	0.875	0.907 +- 0.001
MMC	0.855	0.834	0.803	0.851 +- 0.008
RCA	0.881	0.865	0.839	0.881 +- 0.002
DEEP	0.996	0.994	0.994	0.991 +- 0.324

**Table 8.6** Indian Pines Hyperspectral Dataset Models and Success Metric Scores

Models	Accuracy Score	F1 Score	Kappa Score	Mean and Standard Deviation
LFDA	0.803	0.791	0.771	0.810 +- 0.009
LMNN	0.875	0.835	0.856	0.874 +- 0.002
MLKR	0.773	0.703	0.739	0.774 +- 0.007
NCA	0.772	0.702	0.737	0.764 +- 0.007
ITML	0.628	0.545	0.568	0.602 +- 0.029
LSML	0.773	0.703	0.739	0.765 +- 0.007
MMC	0.754	0.664	0.716	0.743 +- 0.020
RCA	0.783	0.714	0.749	0.775 +- 0.007
DEEP	0.987	0.974	0.984	0.977 +- 1.470



## 9 Conclusion

---

In the project, supervised learning algorithms and weakly supervised learning algorithms, which are metric learning algorithms, and contrastive loss method, which are deep metric learning algorithms, were implemented. Pavia University Dataset and Indian Pines Hyperspectral Datasets were used in the models and a total of 18 models were created. The created models were compared according to overall accuracy, F-Measure and Kappa success value. As a result of the comparisons, it was observed that the deep metric learning approach gave more successful results than supervised learning algorithms and weakly supervised learning algorithms.

## References

---

- [1] X. Cao, Y. Ge, R. Li, J. Zhao, and L. Jiao, "Hyperspectral imagery classification with deep metric learning," *Neurocomputing*, vol. 356, pp. 217–227, 2019.
- [2] Y. Dong, B. Du, L. Zhang, and L. Zhang, "Dimensionality reduction and classification of hyperspectral images using ensemble discriminative local metric learning," *IEEE transactions on geoscience and remote sensing*, vol. 55, no. 5, pp. 2509–2524, 2017.
- [3] E. Pasolli, H. L. Yang, and M. M. Crawford, "Active-metric learning for classification of remotely sensed hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 4, pp. 1925–1939, 2015.
- [4] P. Shippert *et al.*, "Why use hyperspectral imagery?" *Photogrammetric engineering and remote sensing*, vol. 70, no. 4, pp. 377–396, 2004.
- [5] R. Smith, "Introduction to remote sensing of the environment," 2001b. [www.microimages.com](http://www.microimages.com), 2001.
- [6] T. Lillesand, R. W. Kiefer, and J. Chipman, *Remote sensing and image interpretation*. John Wiley & Sons, 2015.
- [7] B. Lu, P. D. Dao, J. Liu, Y. He, and J. Shang, "Recent advances of hyperspectral imaging technology and applications in agriculture," *Remote Sensing*, vol. 12, no. 16, p. 2659, 2020.
- [8] A. Guha, "Mineral exploration using hyperspectral data," in *Hyperspectral Remote Sensing*, Elsevier, 2020, pp. 293–318.
- [9] X. Briottet *et al.*, "Military applications of hyperspectral imagery," in *Targets and backgrounds XII: Characterization and representation*, SPIE, vol. 6239, 2006, pp. 82–89.
- [10] A. A. Gowen, J. Burger, D. O'callaghan, and C. O'donnell, "Potential applications of hyperspectral imaging for quality control in dairy foods," in *1st international workshop on computer image analysis in agriculture, Potsdam, Germany*, 2009.
- [11] M. A. Calin, S. V. Parasca, D. Savastru, and D. Manea, "Hyperspectral imaging in the medical field: Present and future," *Applied Spectroscopy Reviews*, vol. 49, no. 6, pp. 435–447, 2014.
- [12] C. Cucci *et al.*, "Remote-sensing hyperspectral imaging for applications in archaeological areas: Non-invasive investigations on wall paintings and on mural inscriptions in the pompeii site," *Microchemical Journal*, vol. 158, p. 105 082, 2020.
- [13] M. E. Pawlowski, J. G. Dwight, T.-U. Nguyen, and T. S. Tkaczyk, "High performance image mapping spectrometer (ims) for snapshot hyperspectral imaging applications," *Optics Express*, vol. 27, no. 2, pp. 1597–1612, 2019.

- [14] Y. Liang, X. Zhao, A. J. Guo, and F. Zhu, "Hyperspectral image classification with deep metric learning and conditional random field," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 6, pp. 1042–1046, 2019.
- [15] Y. Chen, X. Zhao, and X. Jia, "Spectral–spatial classification of hyperspectral data based on deep belief network," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 8, no. 6, pp. 2381–2392, 2015.
- [16] W. de Vazelhes, C. Carey, Y. Tang, N. Vauquier, and A. Bellet, "Metric-learn: Metric Learning Algorithms in Python," *Journal of Machine Learning Research*, vol. 21, no. 138, pp. 1–6, 2020.
- [17] C. Kha Vu. "Deep metric learning: A (long) survey." (2021), [Online]. Available: <https://hav4ik.github.io/articles/deep-metric-learning-survey>.

## Curriculum Vitae

---

### FIRST MEMBER

**Name-Surname:** Ömer Talha BAYSAN  
**Birthdate and Place of Birth:** 08.03.2000, Eskişehir  
**E-mail:** talha.baysan@std.yildiz.edu.tr  
**Phone:** 0555 056 51 26  
**Practical Training:** CRS Soft Company Software Department

### SECOND MEMBER

**Name-Surname:** Tolga SAĞLAM  
**Birthdate and Place of Birth:** 20.12.1999, İstanbul  
**E-mail:** tolga.saglam@std.yildiz.edu.tr  
**Phone:** 0542 674 42 08  
**Practical Training:** Kalyon Holding Software Department

### Project System Informations

**System and Software:** Windows Operating System, Python  
**Required RAM:** 30GB  
**Required Disk:** 6GB