Turkish Journal of Remote Sensing and GIS
Türk Uzaktan Algılama ve CBS Dergisi

Turk J Remote Sens GIS, March 2024, 5(1): 54–68
Journal homepage: http://www.dergipark.org.tr/en/pub/rsgis

Hyperspectral Image Classification Using Improved Active Deep Learning Framework Including Edge Preserving Filters

Kenar Koruyan Filtreler İçeren Geliştirilmiş Aktif Derin Öğrenme Çerçevesini Kullanan Hiperspektral Görüntü Sınıflandırılması

Zainab Dheyaa AL-Sammarraie[1]*, Ali Can Karaca[1]

[1]*Yildiz Technical University, Electrical-Electronics Faculty, Department of Computer Engineering, 34100, Istanbul/Turkey.*

ORIGINAL PAPER

***Corresponding author:**
Zainab Dheyaa AL-Sammarraie
zainabdh8219@gmail.com

doi: 10.48123/rsgis.1402066

Article history:
Received: 08.12.2023
Accepted: 04.03.2024
Published: 28.03.2024

Abstract

To extract valuable information from satellite data for applications such as agriculture, geological research, and environmental monitoring, the classification of hyperspectral images is an essential task. Labeling each pixel in this process is time-consuming and requires financial resources. To this end, working with a small number of samples is very important. In order to provide high classification performances with a limited number of samples, this paper aims to enhance the performance with an active learning framework. The framework incorporates dimensionality reduction, an edge-preserving filter, and active learning steps. From this perspective, we investigated different edge-preserving filter methods to analyze the effects on performance. By combining edge-preserving filters with dimensionality reduction, the study presents a unique method that improves classification performance while maintaining image quality and reducing noise. The following five edge-preserving smoothing filters are evaluated: weighted least squares (WLS), Joint-Histogram weighted median filter (Joint WMF), fast global image smoother (FGS), bilateral filter (BF), and static/dynamic (SD). Our experiments demonstrate that compared to the reference research (CNN+AL+MRF), the proposed framework increased overall and average accuracies about 2-5% for Indian Pines, Pavia University, and Salinas datasets.

Keywords: *Active learning, Edge preserving filters, Hyperspectral imaging*

Özet

Tarım, jeolojik araştırma ve çevresel izleme gibi uygulamalar için uydu verilerinden değerli bilgiler elde etmek amacıyla hiperspektral görüntünün sınıflandırılması önemli bir görevdir. Bu süreçte her pikselin etiketlenmesi zaman alıcıdır ve mali kaynak gerektirmektedir. Bu amaçla az sayıda örnekle çalışmak çok önemlidir. Sınırlı sayıda örnek altında yüksek sınıflandırma performansı sağlamak için bu makale, performansı aktif bir öğrenme çerçevesiyle geliştirmeyi amaçlamaktadır. Çerçeve, boyut azaltma, kenar koruma filtresi ve aktif öğrenme adımlarını içermektedir. Bu açıdan bakıldığında performans üzerindeki etkilerini analiz etmek için farklı kenar koruyucu filtre yöntemlerini araştırılmıştır. Kenar koruyucu filtreleri boyut azaltmayla birleştiren çalışma, görüntü kalitesini korurken ve gürültüyü azaltırken sınıflandırma performansını artıran benzersiz bir yöntem sunmaktadır. Ağırlıklı En Küçük Kareler (WLS), Ortak Histogram Ağırlıklı Ortanca Filtre (Joint WMF), Hızlı Global Görüntü Yumuşatma (FGS), Bilateral Filtre (BF), and Static/dynamic (SD) olmak üzere toplam beş kenar koruyan filtre değerlendirilmiştir. Deneylerimiz referans araştırmayla (CNN+AL) karşılaştırıldığında önerilen çerçevenin Indian Pines, Pavia Üniversitesi ve Salinas veri kümeleri için genel ve ortalama doğruluğu yaklaşık %2-5 artırdığını göstermektedir.

Anahtar kelimeler: *Aktif öğrenme, Kenar koruyan filtreler, Hiperspektral görüntüleme*

1. Introduction

Classification of hyperspectral images involves identifying various forms of information on Earth's surface using hyperspectral satellite data (Wang et al., 2021). This is achieved through the classification of satellite images, utilizing machine learning and deep learning techniques to identify areas containing various types of living organisms and objects (He et al., 2017). High-accurate hyperspectral image classification requires a comprehensive understanding of the electromagnetic spectrum and its relationship to living beings and objects on Earth's surface (Hong et al., 2019). This data is utilized to generate precise maps of the environment, soil, forests, agriculture, nature, water, and other resources (Alcolea et al., 2020; Hong et al., 2019). These maps play a vital role in scientific and commercial applications, including geological research, natural resource management, and environmental, medical, and agricultural investigations.

The hyperspectral image (HSI) is a crucial type of remote sensing image where each pixel possesses a unique spectral signature, allowing the detection of terrestrial objects invisible to the naked eye. Over the past decade, hyperspectral image analysis has become one of the most sophisticated and rapidly advancing technologies in remote sensing. Thanks to advancements in hyperspectral imaging technology, hyperspectral sensors can now capture substantial data across a wide range of electromagnetic spectrum bands. HSI classification, which is a challenging task in hyperspectral remote sensing (Chen et al., 2018), utilizes a designated discriminant function and a defined set of spectral and spatial image attributes. The process entails the allocation of a distinctive label to each pixel vector. The challenge in HSI arises mainly from the time-consuming and expensive nature of the data annotation, which frequently calls for specific knowledge that may not be easily accessible.

To address the cost and limitations of manual labeling, a solution is proposed, leveraging relevant unlabeled data and employing a convolutional neural network (CNN) to enhance classification performance by extracting crucial spectral-spatial properties (Hu et al., 2020). Image identification and image segmentation extensively employ CNNs, considering the spatial correlation between pixels. By strategically incorporating unlabeled training samples, this technique improves robustness against overfitting with minimal labeled data and enhances generalization capabilities by merging spatial and spectral information from the original hyperspectral image (Haut et al., 2018). Active learning (AL) is a method of machine learning in which the algorithm chooses which data sample to label from a pool of data that has not been labeled. The algorithm selects the samples that are the most informative to label in an iterative manner. The goal of the algorithm is to maximize learning efficiency by concentrating on the data points that are most useful. Because of this, Active learning assumes that certain training samples are more valuable than others when it comes to improving the performance of the classifier system.

By carefully choosing which samples to annotate, AL dramatically lowers the required number of labeled samples. The implementation of active learning aims to select the uncertain samples and ask the expert repeatedly during the iteration, resulting in increasing the number of samples for each round. It improves the effectiveness and affordability of active learning for hyperspectral image categorization (Joshi et al., 2009). Numerous efforts have been made to integrate the classification of HSI with deep learning and AL (Liu et al., 2016), To mitigate the impact of this problem, spatial information should be taken into consideration (Li, 2015). For example, (Kang et al., 2013) applied edge-preserving filtering (EPF) to the postprocessing of classification maps. Propose (Zhang et al., 2019) a model that uses CNN to classify the HSI image and maintain edge information in the picture. Although all these techniques achieve good accuracy, there are still certain issues with them. Chen et al. (2016) employed 1 dimensional (1D), 2 dimensional (2D), and 3 dimensional (3D) CNN for the classification of hyperspectral images. In another work, (Santara et al., 2017) proposed a neural network architecture for spectral and spatial feature learning that is band adaptive.

Principal Component Analysis (PCA) and Minimum Noise Fraction (MNF) have proven to enhance image quality and extract valuable information. Additionally, a trilateral filter has been employed for edge information extraction, contributing to improved model accuracy without introducing complexity; this is achieved by leveraging edge information to distinguish between distinct objects (Gupta et al., 2020).

In a method based on the Bilateral Filter (BF) within the multispectral domain (MBF), PCA is utilized to extract spatial information (Hu et al., 2021). Subsequently, edge-preserving filter techniques in the multispectral domain are applied to filter spectral information after feature extraction (Hu et al., 2021). Utilized a graph cut algorithm to address the labeling problem on Markov Random Field (MRF), constructed on the image grid (Jia et al., 2015). Approach (Kang et al., 2013) prioritizes local optimization of pixel classification maps, placing more emphasis on spectral information than spatial context. The significant advantage lies in preserving the original appearance of probability maps, ensuring the retention of pixel-related information. The proposed method introduces an edge-preserving image smoothing benchmark capable of producing competitive results across a diverse range of image contents (Zhu et al., 2019). This benchmark (Zhu et al., 2019), supported by real image smoothing results within an accompanying image dataset, undergoes verification for contrast enhancement tasks through additional related work.

CNN+AL+MRF method uses the full spectrum method without applying additional preprocessing steps (Cao et al., 2020). In our work, we extend this methodology by introducing a new framework that includes preprocessing steps as dimensionality reduction techniques and edge-preserving techniques as step-by-step spatial information, which is considered preprocessing for HSI classification. Not only does it enable increased complexity during the training process, but it also delivers outstanding classification performance and keeps it sharp and unstabilized. In the smoothing/pruning process, edge-preserving filters preserve edge information. In other words, edge-preserving filters smooth the image while reducing edge blurring effects such as halos and so on. Moreover, they are non-linear in nature. Examples of edge-preserving anti-aliasing filter techniques are WLS, JointWMF, FGS, BF, and SD. This optimization not only reduces complexity during training but also improves feature extraction by incorporating spatial neighborhoods with edge-preserving filters. As a result, this family of filters is very useful in reducing image noise, which makes it highly sought after in computer vision and increases the quality and accuracy of classification. The five edge-preserving smoothing filter techniques:

1. Weighted least square (WLS)
2. JointWMF - Joint-Histogram Weighted Median Filter
3. Fast global image smoother (FGS)
4. Bilateral Filter (BF)
5. Static/Dynamic (SD)

The objective of the experiments was to evaluate the effectiveness of applying dimensionality reduction (DR) processes and edge-preserving filter techniques to enhance accuracy. The combination of both DR and these techniques was hypothesized to yield superior results compared to alternative approaches. Our approach involved employing image processing techniques using five smoothing-based filters. The study aims to achieve dimensionality reduction and eliminate noise from smooth and deformed images or textures while preserving sharp edges. Utilize image processing to enhance or manipulate images, striking a balance between preserving sharp edges and managing noise or texture. Additionally, we sought to investigate whether Markov Random Fields (MRF) improves the overall quality and relevance of the final product to the assigned task. Our method includes a comparison of processing times among various approaches, with a specific emphasis on computational efficiency. The primary focus is to compare the results of overall accuracy (OA), average accuracy (AA), and stratified accuracy of PCA and averaging (AVG) processes with the five filter techniques. Implementation times are also considered to analyze the impact of dimensionality reduction and edge-preserving filtering on improving classification efficiency.

2. Methodology

A block diagram of the proposed active learning framework is given in Figure 1. CNN+AL method first chooses very small samples that are limited in number and reduces the dimension of bands by using either AVG or PCA with the five edge-preserving filter techniques. In order to observe the effectiveness of the proposed framework, we used three datasets, namely, Indian Pines, Pavia University, and Salinas. In literature, edge-preserving filtering is considered a form of post-processing (Wan & Chen 2023), but we use edge-preserving filtering as pre-processing. Its purpose is to reduce noise and smooth an image while preserving fine edges, details, and features. These filters are especially useful when you want to improve image quality by removing noise or reducing unwanted artifacts while maintaining the sharpness and clarity of the boundaries of objects, textures, and structures. The most well-known edge-preserving filters are:

1. Weighted Least Square (WLS) Filter is a popular edge-preserving image smoother that is particularly useful for detail enhancement and HDR tone mapping. Nevertheless, it has a high computational cost and a limited ability to preserve edges (Yang et al. 2024). The WLS framework's current deep learning-based filters are primarily supervised learning-based. They raise productivity but not quality (Yang et al., 2024).
2. Joint-Histogram Weighted Median (JointWMF) Filter is a filter technique that performs weighted median filtering on both the input image and its related histogram. It's useful for preserving image edges and structures. And eliminating outlier noise in data matrices, especially if a high percentage of the matrix points are contaminated with outlier noise (Kilik, 2021).
3. Fast Global Image Smoother (FGS) Filter is an efficient edge-preserving smoothing method based on the WLS formulation called Fast Global Smoother. The linear system has an inhomogeneous Laplacian matrix (Min et al., 2014). This technique is for smoothing images globally. It is a two-stage procedure that consists of a corrective step after the first smoothing run.
4. Bilateral Filter (BF) By modifying the bilateral filter parameters, <text> provides flexibility in the fusion process. With the bilateral filter being a noniterative filter and an efficient implementation scheme being available, the fusion process is faster and computationally simple (Kotwal & Chaudhuri, 2010). It smooths an image while maintaining edges and small features by considering both spatial and intensity similarity.

5. Static/Dynamic (SD) Filtering combines responsive parameter adjustments based on changes in input with fixed parameters. Image processing frequently applies this technique to tasks such as feature extraction and noise reduction, aiming to utilize predefined knowledge while maintaining flexibility in response to a variety of input characteristics (Ham et al., 2015).

Dimension reduction methods reduce the number of features by retaining the most informative bands and discarding redundant or less relevant ones. The specific number of bands retained after dimension reduction would depend on factors such as the chosen method, the variance, and the characteristics of the hyperspectral dataset. The number of bands used after dimension reduction may be determined through experimentation or predefined criteria to preserve sufficient information for classification while reducing computational complexity. After dimensionality reduction and edge-preserving filters, an active learning process is started that focuses on uncertainty samples, and according to the type of data that is being utilized, the number of training examples that are selected at random is determined. For instance, when working with Indian Pines data, the training set consists of only 2% of each class, while the remaining classes are used for testing. Following the training phase in the labeled set, assess all of the unlabeled set's pixels and actively choose the most ambiguous pixels using the Best-Versus-Second Best (BvSB) technique. If the sample's BvSB value is low during this phase, it indicates a confused pixel. After labeling, the algorithm adds these pixels to the labeled set and continues iterating until the predetermined number of iterations is reached. In this way, the AL process adds the most uncertain samples round and round until the number of iterations is reached. By giving priority to the samples that contain the most information for labeling, the AL process aims to successfully maximize learning efficiency. The max-pooling layer follows each convolutional layer, with 500 units in the first fully linked layer and K in the second. Because the training data is limited, the augmentation of data involves the application of translation, rotation, flipping, and cropping techniques. As a classifier, we used an eight-layered CNN architecture, which is proposed in (Cao et al., 2020) the CNN network classifies HSI using an 8x8xD patch with 20 filters in the first convolutional layer and 20 filters in the second. In the last stage, MRF is applied similarly to rectify misclassification in the spatial domain (Cao et al., 2020). For more details, step by step explanation of the algorithm is shared in the Algorithm 1.

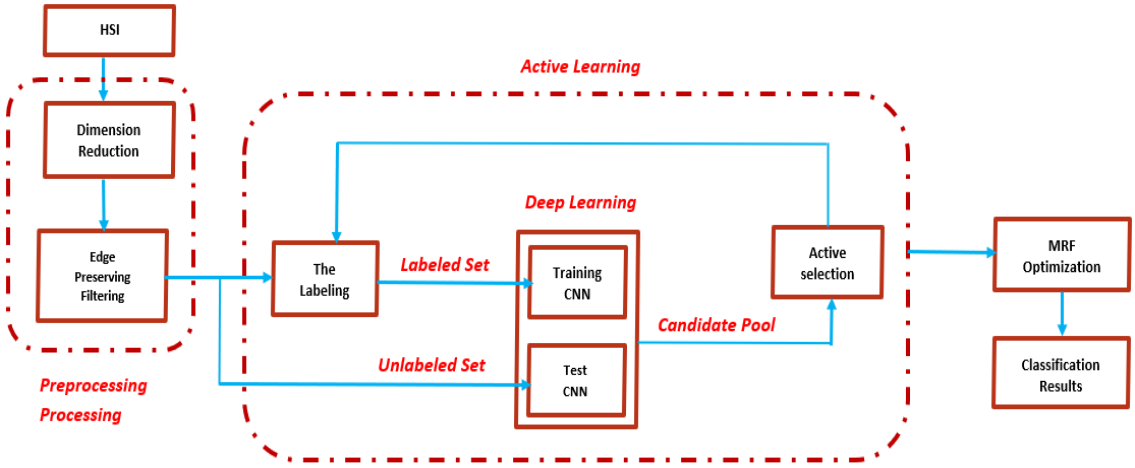

Figure 1. Block diagram of the proposed framework including DR, EPF, CNN+AL, and MRF stages

Algorithm I. Algorithm of the Proposed Framework

Input: Number of training samples *D*, unlabeled pixel set U, maximum number of iterations R, number of pixels actively selected in each round B, flag of using MRF, edge-preserving filter type and its parameters, dimensionality reduction method. r is current iteration number.

Initialization: initialize iteration number r as 0.

Main Loop:

While r < R or terminating conditions which are not met:

1. *Dimensionality Reduction*:
- Perform dimensionality reduction method to D and reduce the dimension to DimR.

2. *Augmentation of the Dataset*:
- Create a new training set DA by augmenting D using DimR and D.

3. *Edge-Preserving Filtering*:
- Apply an edge-preserving filter to the hyperspectral image data in D and U. This filter helps enhance image details, preserve sharp edges, and manage noise. You can choose filter parameters (e.g., bilateral filter, guided filter) to suit your needs.

4. *CNN Training/Fine-Tuning*:
- Train from the stratch or fine-tune a CNN using DA.

5. *Definition of Uncertain Samples*:
- Use the trained CNN to estimate class probabilities for pixels in the unlabeled candidate pool U.
- Actively choose the top B informative pixels from U based on class probabilities.

6. *Extend Training Set*:
- Add patches corresponding to adding the chosen B pixels to the existing training set D, using it as the new training set for the following iteration.

7. *Increment Round*:
- Increment r by 1.

8. *MRF Utilization* (*Optional*):
- If Use MRF is false, go to Labeled.

End of Main Loop

This approach is particularly advantageous when dealing with low sample numbers and diverse ranges, minimal memory usage, simplicity, rapid training durations, avoiding the dimensionality curse, and lowering the complexity of hyperspectral data. AVG and PCA play pivotal roles in simplifying the data representation, making it more manageable for subsequent analysis. Averaging helps reduce noise and smooth out variations, contributing to a cleaner dataset. PCA, on the other hand, identifies and retains the most significant features, discarding less relevant information. The integration of these techniques with edge-preserving filtering enhances model performance by emphasizing the essential dimensions of the data while eliminating less critical ones. When coupled these techniques (DR) with Edge-Preserving Filtering, collectively contribute to an optimized and refined dataset, laying a robust foundation for subsequent classification.

The advantages of employing such pre-processing techniques extend beyond improved model performance. They address practical challenges associated with hyperspectral data, ensuring efficient use of computational resources, and mitigating issues related to limited samples. As a result, this approach not only enhances classification accuracy but also offers practical benefits in terms of resource efficiency and effectiveness. The challenge of analyzing hyperspectral data stems from the dimensionality curse. As an initial step prior to classification, researchers conduct dimensionality reduction. AVG decreases dimensionality by simply averaging successive spectral bands. By averaging the band values, this approach reduces the overall number of spectral bands. AVG does a particularly good job of mitigating the influence of noisy or less distinct bands, which may be common in hyperspectral data (Thilagavathi et al., 2021). In this method the spectral bands are divided into a number of sequential groups after providing the low dimensionality of the data set. For example, when the number of bands is 200 and the reduced dimension is 10, the first group will have spectral band indices ranging from 1 to 20. Later, we implement the AVG technique for every group. To examine and reduce the dimensions of the data into a predetermined number of primary components, employed PCA on a 3D data set as the second technique in DR (Hu et al., 2020). PCA is a popular technique that aims to catch the most significant variations in data (Hu et al., 2020). Transforming the original spectral bands creates a new set of uncorrelated variables, effectively reducing the dimensionality of the data through PCA. The edge-preserving filters are an image processing technique that aim to reduce noise and smooth an image while preserving the sharpness of edges and fine details in the image. In other words, it is a filter designed to enhance image quality by removing noise or unwanted artifacts without blurring key features and boundaries present in the image. Additionally, edge preserving filtering approaches increase interclass variances whereas reduces the intra-class variances. This helps to decrease spectral variability in any class. Our comprehensive approach leverages the synergy between DR techniques for example PCA, and AVG and a suite of advanced edge-preserving filters. These filters, including the Weighted Least Squares (WLS) Filter, Joint Histogram Weighted Mean Filtering (JWMF) Filter, Fast Global Smoothing (FGS) Filter, Bilateral Filter, and Static/Dynamic (SD) Filter, play pivotal roles in improving the precision of classification.

The major feature is that significant information and everything that is not necessary or the ineffectual bands are removed, noise is reduced, and hyperspectral images are smoothed while preserving the sharpness of the images. Additionally, it preserves fine details, which generally correspond to class borders in the hyperspectral image classification process. This algorithm is well suited for situations with a small amount of labeled data. Specifically, we use dimensionality reduction techniques such as PCA and AVG along with five different edge-preserving filter methods. In addition to these preprocessing operations, we improve classification performance by including Markov Random Field (MRF) post-processing methods.

3. Experimental Results

To show the effectiveness of the proposed method, three well-known datasets—Indian Pines, Pavia, and Salinas—were used in the hyperspectral image. We measured the effects of various stages from three different perspectives: i) dimensionality reduction techniques, ii) edge preserving filters, and iii) using post processing (MRF).

All different combinations are compared with the original CNN+AL method. In the dimensionality reduction phase, we used AVG, which reduces the number of bands by calculating the average value of the bands. Alternatively, we used PCA to remove all redundant or ineffective ranges. These last two methods are dimensionality reduction techniques utilized to improve the accuracy results and increase the efficiency of the classifier. During all experiments, we used the number of reduced bands as 10 for PCA and AVG methods. Additionally, in the edge-preserving filter phase, we considered different edge-preserving filters, including Weighted Least Squares (WLS), JointWMF, Fast Global Smoothing (FGS), Bilateral Filter (BF), and Static/Dynamic (SD). The study looks at several parameter configurations and scenarios to gauge the quality and accuracy of the classification. Tables display the method's efficiency on a range of data sets. This thorough study contributes to a deeper comprehension of the usefulness and adaptability of the suggested visualization technique in the context of hyperspectral imaging. Third, the use of MRF has proven to be a successful method to improve classification accuracy. Multiple rounds are conducted in the process, starting with the initial training dataset (D) and a collection of unlabeled pixel data (U).

To ensure a clear ablation study, we carefully assign names to the experimental groups. For example, if CNN-AL is used with PCA, in addition to using with MRF (w MRF) or without MRF (wo MRF) in addition to a specified bilateral edge-preserving filter (BF), the name will be “PCA+BL+CNN-AL w/ MRF." Concatenating the abbreviations of the methods in related combinations effectively highlights the experimental setting. To evaluate the proposed approach's effectiveness, we conducted multiple tests on three benchmark datasets sourced from different regions globally. We executed the method on a PC equipped with an NVIDIA GeForce RTX 3060 graphics card and 64 GB of memory, utilizing the MATLAB environment with the MatConvNet package. Assessed the results for each method, including dimension reduction and technique five filters. The following metrics are numerically compared in terms of:

- Overall Accuracy (OA): It measures the general classification performance in %.
- Average Accuracy (AA): It evaluates average of classwise accuracies in %.
- Computation Time (s): Computation time of each method in second.

Figure 2 depicts a detailed visual comparison of the five unique filtering approaches (WLS, JWMF Filter, FGS, BF, and SD) on three distinct datasets. The figure shows five images, each corresponding to a different filter technique, allowing for a thorough evaluation of their separate effects. There are noticeable variances in noise reduction, smoothing, distortion, and feature enhancement. The distinct differences highlight the importance of filter selection and parameter adjustment in determining the visual quality of hyperspectral images. All methods filter the band images and fuse the spatial information for the related band image. Preserving the edges and smoothing the homogenous region reduces the class variability and simply transforms the spectral signatures of any classes into more similar signatures. In general, SD and BF smooth the images more than the other methods.

Note that hyperparameters about both active learning process and different edge preserving filters are shared in Table 1 and Table 2, respectively. More information can be found from the related references (Yang et al., 2024; Kilik, 2021; Min et al., 2014; Kotwal & Chaudhuri, 2010; Ham et al., 2015).

Table 1. Hyperparameters about active learning during different experiments

Type Data	Learning Rate	Epochs	Number Times Iteration	Number of samples
Indian Pines	0.002	300	5	15
Pavia University	0.001	300	4	5
Salinas	0.002	300	5	15

Table 2. Hyperparameters about different EPF methods

Filter Type	Parameter 1 & Its Value	Parameter 2 & Its Value	Parameter 3 & Its Value
WLS	Smoothness term, 1	Degree of control over the affinities, 1.2	-
JointWMF	Window radius, 10	Standard deviation of kernel, 50	Iteration, 1
FGS	Standard deviation, 0.005	Lambda, 100	-
BF	Window size, 5	Standard deviation, 3	-
SD	Static guadiance, 100	Dynamic guadiance, 400	Step, 3

3.1 Indian Pines

Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor captured the Indian Pines dataset over the Indian Pines test site in northwest Indiana. The dataset consists of a 145 by 145-pixel grid with 220 spectral reflectance bands ranging from 0.4 to 2.5 micrometers.

The environment of Indian Pines has a complex blend of natural perennial flora, cultivated areas, constructed structures, transit infrastructure, and residential areas. Given the mismatch in sample numbers, 16 separate classes pose a unique categorization difficulty. We randomly pick 2% of the samples from each class for training and the rest part of the data is determined as test set. Note that these samples are only for the beginning stage of AL. For each iteration, actively selected 15 samples are added to the training set as is seen from Table 3. Due to randomness, we repeated the experiments for five rounds and the averages of the scores are shared during the experiments. The number of training and test samples are shared in Table 3.

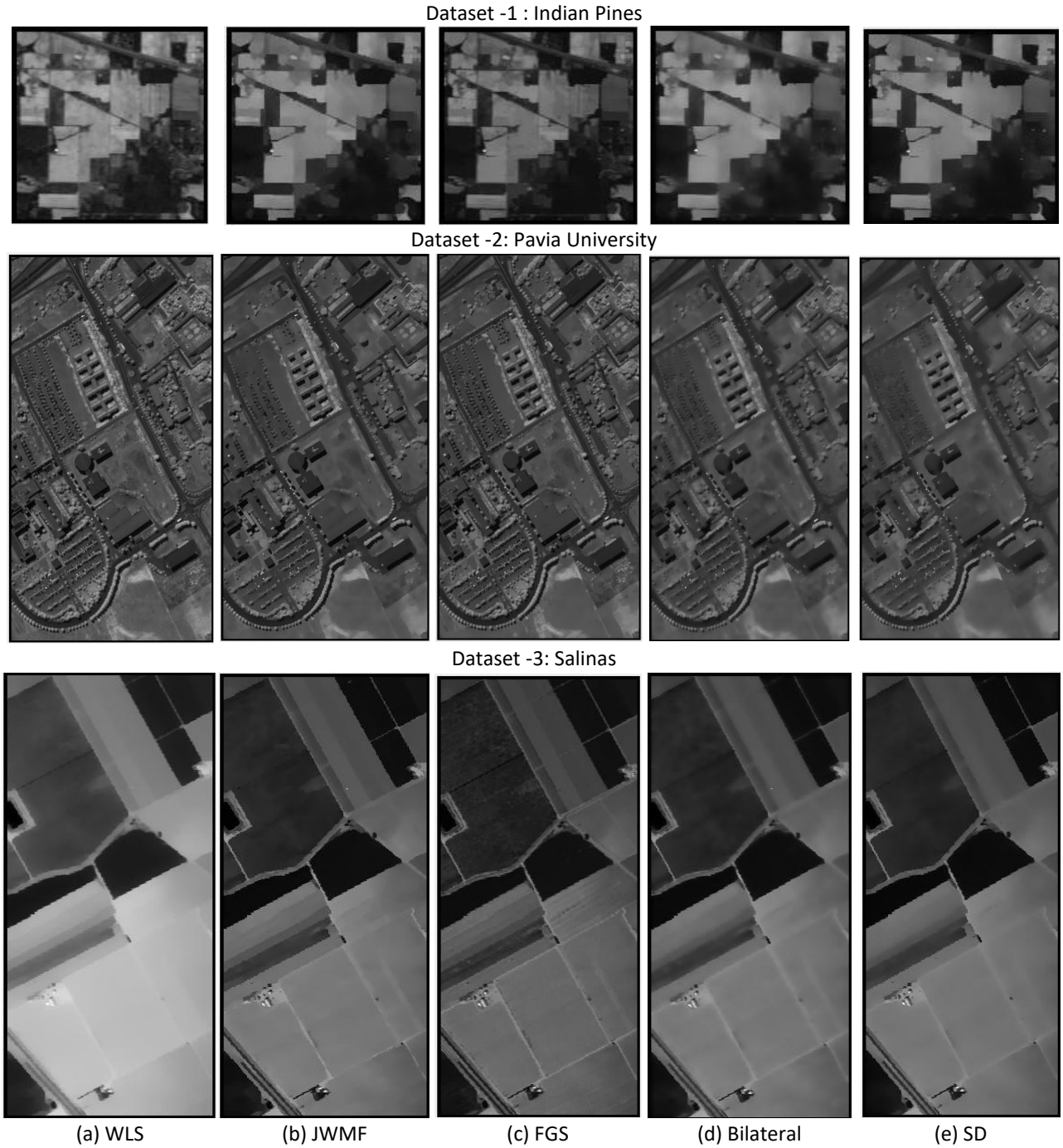

Figure 2. The images show the effect of the filter techniques on the Data Sets with Five filter techniques

Table 3. Number of test and train samples (2%) in Indian Pines (IP) dataset

No	Classes	Train	Test	Total Samples
1	**Alfalfa**	1	45	46
2	**Corn-notill**	29	1399	1428
3	**Corn-mintill**	17	813	830
4	**Corn**	5	232	237
5	**Grass-pasture**	10	473	483
6	**Grass-trees**	15	715	730
7	**Grass-pasture-mowed**	1	27	28
8	**Hay-windrowed**	10	468	478
9	**Oats**	1	19	20
10	**Soybean-notill**	20	952	972
11	**Soybean-mintill**	50	2405	2455
12	**Soybean-clean**	12	581	593
13	**Wheat**	4	201	205
14	**Woods**	26	1239	1265
15	**Buildings-Grass-Trees**	8	378	386
16	**Stone-Steel-Towers**	2	91	93
	Total	**211**	**10038**	**10249**

The initial training round has 300 epochs, and future rounds have the same length. The selection of 15 samples aids each round for training. Our system is based on a comprehensive pipeline that includes dimensionality reduction techniques as well as edge-preserving filter techniques. In Table 4, we shared the classification accuracy and computation times of the methods using with and without MRF and different edge-preserving filters (WLS, JWMF, FGS, BF, and SD). We also give the results of the original CNN+AL method in the first row. Note that this table does not include any dimension reduction algorithm to measure the effect of edge-preserving filters. It is seen that applying edge-preserving filters generally result in higher overall accuracy than the original version. However, WLS gives lower AA values than the original one when MRF is active. FGS, BF, and SD filters achieved a higher than AA score of 95% both with and without MRF versions. The highest accuracy was measured for SD+CNN+AL, with scores of OA 99.00% and AA 98.62%. Table 5 includes the AVG dimensionality reduction method, while Table 6 includes the PCA dimensionality reduction method. Here, it is seen that using dimensionality reduction has a significant impact. AVG and PCA methods use the five-filter technique, and the effect of this combination is noticeably clear on the classification results, which are measured numerically by overall accuracy (OA), average accuracy (AA), and execution or computation time. Thus, all the filters exceeded the original CNN+AL performance, and SD is the best technique that gives a high accuracy result in these three tables, as AVG+CNN+AL and PCA+CNN+AL obtain OA scores of 99.03 and 99.02, respectively.

Table 4. Classification performances (%) and computation times of the methods without DR for IP

Classes	CNN+AL		CNN+AL+ WLS Filter		CNN+AL+ JWMF Filter		CNN+AL + FGS Filter		CNN+AL + Bilateral Filter		CNN+AL + SD Filter	
	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF
Alfalfa	97.43	93.69	98.34	92.37	97.55	90.55	97.67	97.16	97.65	81.34	97.67	85.41
Corn-notill	98.15	81.03	98.27	87.08	98.13	82.45	98.11	96.97	98.35	98.61	98.47	95.03
Corn-mintill	99.56	93.83	100.00	71.62	100.00	67.70	100.00	100.00	100.00	92.73	100.00	96.33
Corn	98.14	92.36	98.55	93.61	96.88	93.37	96.88	100.00	96.88	97.09	97.48	96.86
Grass-pasture	99.04	97.93	98.90	96.70	99.31	97.79	99.31	99.17	99.31	99.72	99.31	99.45
Grasstrees	100.00	98.05	100.00	99.16	100.00	97.89	96.88	100.00	100.00	97.22	100.00	99.79
Grass-pasture-mowed	97.86	88.42	98.19	87.06	98.69	85.56	98.81	98.05	98.72	98.07	98.81	97.07
Hay-windrowed	97.61	94.52	99.38	92.09	99.58	92.12	99.53	97.61	99.57	96.30	99.58	95.36
Oats	96.47	92.06	98.65	90.88	99.14	85.88	98.80	96.58	98.81	96.08	98.80	95.38
Soybean-notill	99.52	99.52	100.00	94.20	100.00	96.17	100.00	99.51	100.00	99.52	100.00	99.02
Soybean-mintill	99.60	99.44	99.84	99.04	100.00	99.04	100.00	99.60	100.00	99.68	100.00	99.60
Soybean-clean	98.65	79.45	100.00	70.03	100.00	74.05	100.00	99.16	100.00	96.91	100.00	97.73
Wheat	81.72	80.65	85.87	98.91	87.64	93.55	91.11	82.22	90.00	88.89	89.89	91.01
Woods	96.00	67.31	98.08	73.08	98.00	73.08	97.96	93.88	97.96	89.80	97.78	88.89
Buildings-Grass-Trees	76.00	92.00	73.08	0.00	100.00	0.00	100.00	78.26	100.00	36.00	100.00	64.00
Stone-Steel-Towers	89.47	91.99	60.00	20.00	100.00	36.84	100.00	100.00	100.00	100.00	100.00	89.47
Overall Accuracy (OA)	**95.33**	**92.70**	**98.78**	**91.17**	**98.92**	**90.19**	**98.94**	**97.93**	**98.95**	**95.11**	**99.00**	**95.26**
Average Accuracy (AA)	**95.00**	**90.15**	**94.20**	**89.11**	**98.43**	**89.13**	**98.64**	**96.02**	**98.58**	**91.75**	**98.61**	**93.15**
Computation Time (s)	**252.54**	**166.64**	**196.22**	**194.63**	**195.37**	**189.10**	**192.78**	**189.27**	**202.66**	**200.03**	**210.95**	**202.98**

Table 5. Classification performances (%) and computation times of the methods with averaging for IP

Classes	CNN+AL + AVG		CNN+AL + AVG+WLS Filter		CNN+AL + AVG+JWMF Filter		CNN+AL + AVG+FGS Filter		CNN+AL + AVG+ Bilateral Filter		CNN+AL + AVG+SD Filter	
	With MRF	With out MRF	With MRF	With out MRF	With MRF	Without MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF
Alfalfa	97.64	97.42	97.64	96.35	97.20	92.33	97.71	97.57	97.62	85.58	97.42	87.08
Corn-notill	98.64	97.40	98.64	97.27	93.18	92.44	98.64	97.29	98.64	98.27	98.64	94.20
Corn-mintill	100.00	100.00	100.00	99.56	100.00	97.3	100.00	100.00	100.00	93.01	100.00	95.61
Corn	98.97	98.14	98.76	98.35	91.30	98.14	99.18	98.14	98.96	97.72	98.76	96.4
Grass-pasture	99.32	99.04	99.31	99.31	99.72	99.31	99.45	99.03	99.4	99.72	99.31	99.45
Grasstrees	100.00	100.00	100.00	100.00	100.00	99.79	100.00	100.00	100.00	97.06	100.00	99.79
Grass-pasture-mowed	98.41	97.99	98.40	98.61	98.72	97.86	98.49	97.96	98.61	98.40	98.40	97.65
Hay-windrowed	99.25	98.13	99.58	98.87	99.62	94.69	99.25	97.74	99.54	94.64	99.75	93.38
Oats	98.83	96.33	99.15	96.11	98.31	95.28	98.82	96.64	98.65	95.79	98.89	95.44
Soybean-notill	100.00	99.03	100.00	99.03	100.00	99.52	100.00	99.03	100.00	99.52	100.00	99.04
Soybean-mintill	100.00	99.68	100.00	99.60	100.00	99.68	100.00	99.60	100.00	99.44	100.00	99.12
Soybean-clean	100.00	98.65	100.00	97.55	100.00	99.46	100.00	98.65	100.00	97.30	100.00	98.64
Wheat	88.17	81.74	89.01	83.52	89.25	73.12	88.17	81.72	88.17	87.10	89.01	86.81
Woods	98.11	98.11	98.08	98.08	96.00	90.00	98.04	96.08	98.00	86.00	98.04	90.20
Buildings-Grasstrees	75.00	83.33	73.08	76.92	100.00	66.67	72.00	76.00	73.08	38.46	100.00	61.54
Stone-Steel-Towers	100.00	95.00	100.00	100.00	100.00	100.00	100.00	90.00	100.00	100.00	100.00	90.00
Overall Accuracy (OA)	**98.09**	**96.25**	**98.98**	**98.13**	**98.20**	**95.93**	**98.92**	**97.97**	**98.97**	**95.25**	**99.03**	**94.90**
Average Accuracy (AA)	**96.25**	**94.21**	**96.98**	**96.20**	**97.71**	**93.48**	**96.86**	**95.34**	**96.92**	**91.75**	**98.64**	**92.78**
Computation Time (s)	**165.46**	**161.04**	**179.53**	**114.52**	**188.95**	**152.53**	**204.74**	**189.99**	**205.78**	**190.46**	**202.78**	**190.47**

Table 6. Classification performances (%) and computation times of the methods with PCA for IP

Classes	CNN+AL + PCA		CNN+AL + PCA+WLS Filter		CNN+AL + PCA+JWMF Filter		CNN+AL + PCA+FGS Filter		CNN+AL + PCA+ Bilateral Filter		CNN+AL + PCA+SD Filter	
	With MRF	With out MRF	With MRF	With out MRF	With MRF	Without MRF	With MRF	Without MRF	With MRF	With out MRF	With MRF	With out MRF
Alfalfa	97.43	93.36	97.64	96.49	97.64	92.05	97.63	97.35	97.63	85.40	97.63	89.81
Corn-notill	98.15	83.50	98.15	98.03	98.27	91.46	98.15	97.65	98.15	98.02	98.15	94.31
Corn-mintill	99.56	77.33	100.00	99.13	100.00	97.39	100.00	100.00	100.00	93.39	100.00	96.51
Corn	98.14	93.35	96.89	98.34	96.90	98.35	96.90	98.14	96.90	97.52	97.70	97.08
Grass-pasture	99.04	96.28	99.31	99.45	99.31	99.31	99.31	99.17	99.31	99.7	99.31	99.45
Grasstrees	100.00	98.95	100.00	100.00	100.00	99.58	100.00	100.00	100.00	97.27	100.00	100.00
Grass-pasture-mowed	97.86	84.53	98.71	98.29	98.72	97.87	98.71	97.96	98.72	98.40	98.72	97.54
Hay-windrowed	97.61	91.41	99.54	98.58	99.54	94.93	99.54	97.49	99.58	94.78	99.54	92.47
Oats	96.47	89.04	98.82	95.78	98.82	94.94	98.82	96.63	98.82	95.79	98.82	94.60
Soybean-notill	99.52	94.23	100.00	99.04	100.00	994.94	100.00	100.00	100.00	99.52	100.00	99.04
Soybean-mintill	99.60	98.81	100.00	99.52	100.0	99.52%	100.00	99.68	100.00	99.44	100.00	98.96
Soybean-clean	98.65	74.86	100.00	97.83	100.00	98.65%	100.00	98.64	100.00	97.57	100.00	98.92
Wheat	81.72	96.74	89.13	83.70	88.17	73.12	88.17	81.72	88.17	88.17	90.04	85.87
Woods	96.00	74.51	98.00	98.00	98.04	96.08	98.04	98.04	98.00	86.00	97.96	87.76
Buildings-Grass-Trees	76.00	0.00	100.00	76.00	100.00	69.23	100.00	76.92	100.00	38.46	100.00	73.08
Stone-Steel-Towers	89.47	26.32	100.00	95.00	100.00	100.00	100.00	95.00	100.00	100.00	100.00	90.00
Overall Accuracy (OA)	**97.97**	**90.72**	**98.92**	**98.08**	**98.92**	**95.82**	**98.91**	**95.90**	**98.92**	**93.29**	**99.02**	**95.07**
Average Accuracy (AA)	**95.33**	**88.91**	**98.51**	**95.82**	**98.46**	**93.84**	**98.46**	**94.83**	**98.46**	**91.84**	**98.62**	**93.46**
Computation Time (s)	**151.76**	**144.40**	**179.46**	**177.59**	**193.35**	**192.70**	**190.77**	**189.96**	**199.90**	**198.57**	**203.90**	**200.02**

Note that, among all the results, the accuracy percentages show how well different techniques can classify several types of land cover. For example, some categories, such as "grass trees" and “soybeans,” achieved close to 100% accuracy, while others, such as "wheat" and “buildings, grass, and trees,” were less accurate, which showed a variance in data classification performance. But in general, we conclude that the best result and the shortest time were obtained using the technique (CNN + AL + AVG + SD w/ MRF), where (OA = 99.03, AA = 98.64). In general, using dimensionality reduction methods not only increases performance but also decreases computation time.

3.2 Pavia University

A popular hyperspectral remote sensing dataset for image analysis and classification applications is the University of Pavia dataset. The Imaging Spectrometer with Reflective Optics System, an aerial hyperspectral sensor, gathers the dataset over the Pavia campus of the University of Pavia. As Table 6 illustrates, the Pavia dataset has nine distinct classifications. Its dimensions are 610 pixels wide by 340 pixels high, with 103 spectral bands. Consequently, it is a 3D data cube. These data provide a number of difficulties arising from the high dimensionality of hyperspectral data.

As shown in Table 7, select 1% of the samples from each class. The training runs consist of 300 epochs repeated five times, and we effectively select five samples as configured for the training set.

Table 7. Number of test and train samples (1%) in Pavia University (PU) dataset

No	Classes	Train	Test	Total Samples
1	Asphalt	67	6564	6631
2	Meadows	187	18462	18649
3	Gravel	21	2078	2099
4	Trees	31	3033	3064
5	Painted metal sheets	14	1331	1345
6	Bare Soil	51	4987	5029
7	Bitumen	14	1316	1330
8	Self-Blocking Bricks	37	3645	3682
9	Shadows	10	947	947
	Total	**432**	**42344**	**42776**

Table 8, Table 9, and Table 10 show the experimental results of the five filtering techniques, with the same order of Indian Pines results. When the dimensionality reduction method is not used in the framework, applying an edge-preserving filter generally increases overall accuracy, and average accuracy scores for MRF preprocessing are active. Using MRF as a postprocess always provides better accuracy because it adapts spatial relationships. Note the accuracy results through the OA and AA values of SD filter for all three tables, where the results appeared as follows: OA scores of 94.99, 98.00, and 97.00 and AA scores of 92.25, 92.54, and 91.38 in Table 8, Table 9, and Table 10, respectively. Despite these outcomes, certain classes—such as Self-Blocking bricks—give lower individual results than the remaining classes for the Pavia University data.

According to the results, CNN+AL+AVG+SD Filter w/ MRF technique produced the highest accuracies in terms of classification accuracy, with an OA value of 98.00 and an AA value of 92.54, which is significantly higher than the original CNN+AL method. On the other hand, SD filter has the highest computational complexity at all. In general, we can see that each edge-preserving filtering strategy can assist increase efficiency and achieve great classification, even with the difficulties brought on by the large dimensionality of the hyperspectral data in the Pavia University dataset.

CNN+AL obtains accuracy of "self-blocking bricks" class as 39.56 and 40.24 by applying with and without MRF, respectively. Here, in some DR and EPF combinations such as CNN+AL+AVG+WLS, the relevant classwise accuracy reached to more than 90%. This case is generally seen when AVG and any EPF method is combined. Additionally, AVG achieves the best results compared to PCA and standard results.

Table 8. Classification performances (%) and computation times of the methods without DR for PU

Classes	CNN+AL		CNN+AL+ WLS Filter		CNN+AL+ JWMF Filter		CNN+AL + FGS Filter		CNN+AL + Bilateral Filter		CNN+AL + SD Filter	
	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	Without MRF	With MRF	With out MRF	With MRF	With out MRF
Asphalt	99.26	96.63	99.32	91.78	99.36	80.22	99.36	94.61	99.39	67.01	99.92	52.37
Meadows	94.70	91.28	95.03	99.83	95.17	98.07	94.73	94.71	94.95	98.14	95.12	98.51
Gravel	99.76	98.52	99.96	86.08	99.76	99.67	99.78	96.64	99.78	99.71	99.76	99.87
Trees	97.22	98.01	98.90	85.95	99.04	89.95	97.92	97.29	97.22	84.31	98.01	73.94
Painted metal sheets	99.93	99.93	99.98	99.03	99.93	99.85	99.93	98.21	99.94	99.93	100.00	99.33
Bare Soil	100.0	96.95	100.00	81.58	100.00	94.92	100.00	91.96	100.00	96.70	100.00	96.16
Bitumen	97.52	96.46	98.03	94.73	97.63	96.84	97.59	96.73	98.13	96.70	98.03	74.72
Self-Blocking Bricks	39.56	40.24	42.20	37.88	44.07	40.15	42.66	40.13	41.99	26.11	40.62	23.98
Shadows	98.84	99.26	99.24	98.31	99.24	97.67	98.85	97.36	98.93	96.73	98.83	65.20
Overall Accuracy (OA)	**92.06**	**90.81**	**93.29**	**90.00**	**94.02**	**89.26**	**94.22**	**90.13**	**94.10**	**86.24**	**94.99**	**83.30**
Average Accuracy (AA)	**91.86**	**89.83**	**92.52**	**88.35**	**92.69**	**88.59**	**92.31**	**89.74**	**92.26**	**85.04**	**92.25**	**76.01**
Computation Time (s)	**398.3**	**371.09**	**420.03**	**410.22**	**498.46**	**436.25**	**419.33**	**400.91**	**5817**	**522.80**	**573.13**	**514.97**

Table 9. Classification performances (%) and computation times of the methods with averaging for PU

Classes	CNN+AL + AVG		CNN+AL + AVG+WLS Filter		CNN+AL + AVG+JWMF Filter		CNN+AL + AVG+FGS Filter		CNN+AL + AVG+ Bilateral Filter		CNN+AL + AVG+SD Filter	
	With MRF	With out MRF	With MRF	Without MRF	With MRF	Without MRF	With MRF	Without MRF	With MRF	Without MRF	With MRF	Without MRF
Asphalt	98.73	86.65	98.98	81.27	98.97	83.02	99.30	87.41	98.73	86.03	99.27	94.69
Meadows	99.07	95.31	99.59	98.42	100.00	98.39	99.99	98.33	99.08	97.86	99.98	98.54
Gravel	72.32	68.31	75.21	90.27	80.87	82.64	81.12	85.54	79.24	81.06	80.02	72.84
Trees	93.24	68.93	94.12	89.06	95.02	89.78	94.73	89.19	95.22	91.31	94.60	93.45
Painted metal sheets	99.85	98.88	100.00	97.83	99.99	97.46	100.00	96.79	100.00	98.29	99.99	99.78
Bare Soil	99.12	78.57	99.98	91.94	99.98	92.56	99.12	94.71	99.99	96.71	100.0	87.83
Bitumen	94.80	53.54	95.07	58.14	97.21	83.79	95.99	78.58	95.15	84.05	95.90	58.30
Self-Blocking Bricks	92.00	66.65	93.08	48.11	94.52	70.84	93.21	70.75	93.89	81.07	93.03	88.57
Shadows	62.16	65.22	63.02	97.57	64.96	96.61	66.99	98.31	70.01	97.78	70.10	97.67
Overall Accuracy (OA)	**95.76**	**84.47**	**96.86**	**88.31**	**97.13**	**91.03**	**97.99**	**91.89**	**97.10**	**92.73**	**98.00**	**92.97**
Average Accuracy (AA)	**90.14**	**75.78**	**91.01**	**83.63**	**92.39**	**88.34**	**92.27**	**88.85**	**92.37**	**90.46**	**92.54**	**87.96**
Computation Time (s)	**356.7**	**332.8**	**411.23**	**395.26**	**410.81**	**411.01**	**402.28**	**395.92**	**458.83**	**458.83**	**536.6**	**509.26**

Table 10. Classification performances (%) and computation times of the methods with PCA for PU

Classes	CNN+AL + PCA		CNN+AL + PCA+WLS Filter		CNN+AL + PCA+JWMF Filter		CNN+AL + PCA+FGS Filter		CNN+AL + PCA+ Bilateral Filter		CNN+AL + PCA+SD Filter	
	With MRF	With out MRF	With MRF	With out MRF	With MRF	Without MRF	With MRF	Without MRF	With MRF	Without MRF	With MRF	With out MRF
Asphalt	99.26	85.35	96.32	79.50	97.24	81.27	98.73	84.01	98.73	80.31	98.26	80.32
Meadows	97.58	96.51	99.46	98.90	99.39	98.42	99.07	97.93	99.07	98.27	99.62	99.31
Gravel	99.62	33.35	53.46	48.44	72.71	90.27	72.38	84.98	72.33	71.98	80.88	73.61
Trees	97.88	91.30	88.99	91.35	89.66	89.06	93.24	91.18	93.24	85.78	94.78	87.54
Painted metal sheets	99.78	95.09	100.0	99.85	100.00	97.83	99.85	99.03	99.85	98.30	99.32	97.94
Bare Soil	96.32	87.57	93.57	71.98	95.30	91.94	99.12	94.29	99.12	88.09	99.69	88.41
Bitumen	96.68	76.02	89.15	47.35	92.81	58.14	94.80	63.36	94.80	60.39	94.81	69.44
Self-Blocking Bricks	43.88	78.63	93.95	66.03	78.52	48.11	92.08	60.79	91.99	82.99	92.06	82.59
Shadows	97.99	40.49	98.84	99.26	98.62	97.57	62.22	97.57	62.16	98.96	62.99	98.62
Overall Accuracy (OA)	**93.24**	**86.80**	**94.48**	**85.42**	**95.57**	**88.31**	**95.77**	**89.98**	**95.76**	**90.10**	**97.00**	**90.69**
Average Accuracy (AA)	**92.10**	**76.03**	**90.41**	**78.07**	**91.58**	**83.63**	**90.17**	**85.90**	**90.14**	**85.01**	**91.38**	**86.42**
Computation Time (s)	**350.73**	**319.99**	**429.9**	**410.2**	**403.25**	**382.24**	**375.74**	**331.96**	**487.17**	**441.96.**	**355.72**	**340.0**

3.3 Salinas

224-band AVIRIS sensor was used to obtain this high-spatial-resolution (3.7-meter) picture over the Californian Salinas Valley. The covered region consists of 512 lines, with 217 samples distributed among them. In a similar manner to Indian Pines, the twenty water-absorbing bands were discarded. For this image, only at-sensor radiance measurements were available. Vegetable fields, wine farms, and arid soils are all part of it. The training and testing experiments are performed on Salinas data similar to that in Indian data, with the same scenario in the last data mention set where 2% of samples are selected from each class and with the same training rounds and size, so they are identical to those in Indian data in mechanism. Table 11 exhibits the total volume of training and test samples for every class. The Salinas dataset is distinct from others in its rich spectral information and large dimensions.

The results of the Salinas dataset are shared in Tables 12, 13, and 14 in the same order as the previous experiments. Similar to the previous experiment outputs, using edge-preserving filters increases the effectiveness of the active learning method. When both DRs are coupled with edge-preserving filters, the methods obtain at least an OA score of 98%, which is at least 3% higher than the original CNN+AL. The best result found in Table 10 is CNN + AL + SD w/MRF with an OA score of 97.16 and an AA score of 97.16. The best results found in AVG and PCA are obtained in CNN+AL + FGS w/MRF with OA scores of 98.06 and 99.33, respectively. It is evident that the best dimensionality reduction technique can vary depending on the dataset. Therefore, classification increases its efficiency in several ways and techniques, including filtering techniques that remove noise or texture from the data while preserving sharp edges, thus improving performance.

Table 11. Number of test and train samples (2%) in Salinas (SA) dataset

No	Classes	Train	Test	Total Samples
1	Brocoli green weeds 1	41	1968	2009
2	Brocoli green weeds 2	38	3688	3726
3	Fallow	40	1936	1976
4	Fallow rough plow	28	1366	1394
5	Fallow smooth	54	2624	2678
6	Stubble	80	3879	3959
7	Celery	72	3507	3579
8	Grapes untrained	226	11045	11271
9	Soil vinyard develop	125	6078	6203
10	Corn senesced green weeds	66	3212	3278
11	Lettuce romaine 4wk	22	1046	1068
12	Lettuce romaine 5wk	39	188	1927
13	Lettuce romaine 6wk	19	897	916
14	Lettuce romaine 7wk	22	1048	1070
15	Vinyard untrained	146	7122	7268
16	Vinyard vertical trellis	37	1770	1807
	Total	**1055**	**53074**	**54129**

In contrast, applying edge preservation filters to HSI with many spectral bands is computationally intensive, especially in scenarios where maintaining fine details is critical. This processing requires algorithms for each band or layer of the spectrum. This data is often processed individually, resulting in a significant increase in processing time compared to alternative methods. For example, in the Indian data set, CNN+AL + AVG took 165.46 seconds while CNN+AL + AVG + WLS took 179.53 seconds. So, the balance between computational complexity and the need to maintain an accurate edge remains a challenge in processing spatial spectrum images.

In Salinas dataset, selection of different EPF types does not generally affect on the accuracies compared to Indian Pines and Pavia University datasets. For example, when AVG is used as DR method with MRF, all EPF methods perform nearly 98% score of OAs and AAs which are at least 2% higher than CNN+AL method. However, in without MRF versions, WLS gets the lowest accuracies among these five EPF methods. This shows that MRF is a promising technique and needed to be applied as a post processing step.

Table 12. Classification performances (%) and computation times of the methods without DR for SA

Classes	CNN+AL		CNN+AL+ WLS Filter		CNN+AL+ JWMF Filter		CNN+AL + FGS Filter		CNN+AL + Bilateral Filter		CNN+AL + SD Filter	
	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	Without MRF
Brocoli green weeds 1	99.33	98.82	100.0	98.48	100.0	100.0	100.0	99.95	100.00	98.83	100.0	100.00
Brocoli green weeds 2	99.50	99.75	100.0	99.81	100.0	99.89	100.0	99.89	100.00	99.53	100.0	99.78
Fallow	87.19	86.45	89.09	87.13	86.42	86.50	89.24	93.39	89.82	88.94	90.11	91.78
Fallow rough plow	93.48	93.26	93.48	93.78	94.24	96.19	94.10	94.65	94.21	92.83	94.42	75.39
Fallow smooth	99.37	99.14	99.96	99.81	99.96	99.85	99.96	99.43	99.96	99.69	99.96	99.81
Stubble	98.39	97.99	100.0	99.92	100.0	99.92	100.0	99.95	100.00	99.92	100.0	99.87
Celery	98.21	97.27	99.37	98.20	99.32	98.54	99.37	98.06	99.37	97.38	99.37	98.46
Grapes untrained	91.77	83.75	92.13	82.56	94.04	76.04	91.45	75.98	92.05	76.42	91.97	73.90
Soil vinyard develop	98.19	99.67	99.79	98.92	99.79	99.77	99.79	99.75	99.79	99.77	99.77	99.77
Corn senesced green weeds	96.47	95.20	96.69	95.01	96.66	94.11	96.67	95.08	97.12	94.27	97.23	94.60
Lettuce romaine 4wk	98.22	96.92	100.0	98.18	100.0	98.09	100.0	98.18	100.00	97.80	100.0	96.65
Lettuce romaine 5wk	99.25	98.09	99.95	98.25	100.0	98.09	99.95	97.99	99.95	97.82	100.0	97.67
Lettuce romaine 6wk	97.61	97.77	97.97	97.77	97.66	97.77	98.37	97.66	98.37	98.33	97.66	98.11
Lettuce romaine 7wk	96.66	96.61	98.38	95.04	98.38	97.33	98.37	97.23	98.37	98.38	98.38	96.47
Vinyard untrained	84.29	77.29	87.86	80.38	85.20	91.84	88.18	88.33	88.18	93.33	87.02	93.05
Vinyard vertical trellis	98.02	96.32	98.59	95.81	98.59	96.15	98.47	96.21	98.53	98.02	98.47	96.78
Overall Accuracy (OA)	**95.56**	**92.58**	**97.04**	**93.59**	**96.50**	**93.11**	**97.02**	**93.24**	**97.02**	**94.01**	**97.16**	**92.86**
Average Accuracy (AA)	**96.00**	**94.65**	**97.08**	**94.94**	**96.89**	**95.63**	**97.12**	**95.73**	**97.23**	**95.70**	**97.16**	**94.51**
Computation Time (s)	**869.0**	**869.07**	**1001.1**	**987.57**	**1000**	**971.1**	**959.2**	**935.23**	**1045.70**	**1000.11**	**1089.6**	**1023.56**

Table 13. Classification performances (%) and computation times of the methods with averaging for SA

Classes	CNN+AL + AVG		CNN+AL + AVG+WLS Filter		CNN+AL + AVG+JWMF Filter		CNN+AL + AVG+FGS Filter		CNN+AL + AVG+ Bilateral Filter		CNN+AL + AVG+SD Filter	
	With MRF	With out MRF	With MRF	With out MRF	With MRF	Without MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF
Brocoli green weeds 1	100.00	100.00	100.0	98.20	100.00	97.76	100.00	99.75	100.0	99.95	100.0	99.75
Brocoli green weeds 2	99.88	99.72	100.0	99.97	100.00	99.41	100.00	99.92	100.0	99.52	100.0	96.23
Fallow	100.00	99.95	99.54	99.90	99.64	100.00	99.56	99.80	99.54	100.0	99.54	100.00
Fallow rough plow	99.86	98.99	99.64	83.61	99.55	93.52	99.64	99.86	99.64	99.07	99.64	97.26
Fallow smooth	98.66	97.46	98.65	100.0	98.55	99.96	98.65	99.25	98.65	98.99	98.65	99.59
Stubble	99.95	98.99	99.99	98.65	99.30	98.66	99.21	98.66	99.19	98.53	99.19	98.45
Celery	98.68	98.15	99.78	92.61	99.78	98.26	99.98	99.56	99.78	98.91	99.78	96.64
Grapes untrained	94.95	88.12	96.82	68.11	95.93	84.88	96.84	91.48	96.33	85.01	95.81	37.17
Soil vinyard develop	99.80	99.45	99.79	99.79	99.79	99.68	99.80	99.79	99.79	99.79	99.79	99.79
Corn senesced green weeds	89.63	83.11	98.50	85.11	98.53	95.41	98.72	95.36	98.50	93.82	98.53	94.00
Lettuce romaine 4wk	94.26	89.44	98.11	2.17	98.72	90.57	98.13	94.73	98.11	95.39	98.11	65.82
Lettuce romaine 5wk	95.86	90.92	97.81	97.65	97.71	96.78	97.71	97.24	97.76	94.44	97.71	94.99
Lettuce romaine 6wk	96.68	93.47	96.61	98.14	96.61	97.59	96.60	96.38	97.20	97.59	96.60	97.26
Lettuce romaine 7wk	97.28	95.50	97.38	97.85	97.38	90.63	97.47	97.66	97.38	96.25	97.38	92.96
Vinyard untrained	89.92	78.00	87.48	96.46	88.39	97.90	88.36	84.51	88.40	93.03	89.32	97.10
Vinyard vertical trellis	99.06	97.39	98.23	98.28	98.28	94.23	98.23	93.95	98.23	95.34	98.22	95.11
Overall Accuracy (OA)	**97.06**	**94.28**	**98.01**	**88.34**	**98.01**	**95.82**	**98.05**	**96.53**	**98.01**	**95.39**	**98.00**	**91.21**
Average Accuracy (AA)	**97.15**	**94.29**	**98.02**	**88.53**	**98.01**	**95.95**	**98.06**	**96.74**	**98.03**	**96.60**	**98.02**	**91.38**
Computation Time (s)	**859.63**	**853.10**	**885.6**	**870.0**	**934.59**	**984.26**	**842.73**	**803.2**	**848.5**	**827.2**	**840.1**	**823.95**

Table 14. Classification performances (%) and computation times of the methods with PCA for SA

Classes	CNN+AL + PCA		CNN+AL + PCA+WLS Filter		CNN+AL + PCA+JWMF Filter		CNN+AL + PCA+FGS Filter		CNN+AL + PCA+ Bilateral Filter		CNN+AL + PCA+SD Filter	
	With MRF	With out MRF	With MRF	Without MRF	With MRF	Without MRF	With MRF	With out MRF	With MRF	With out MRF	With MRF	With out MRF
Brocoli green weeds 1	100.00	99.65	100.0	100.00	100.00	99.25	100.00	100.0	100.0	100.0	100.0	100.0
Brocoli green weeds 2	98.81	99.87	100.0	99.97	100.00	99.46	100.00	100.0	100.0	99.89	100.0	99.78
Fallow	99.23	100.00	100.0	100.00	100.00	100.00	100.00	99.95	100.0	100.0	100.0	100.0
Fallow rough plow	98.16	98.16	99.35	80.61	99.42	92.41	99.35	99.35	99.35	99.06	99.35	96.39
Fallow smooth	98.25	98.34	99.17	100.00	99.32	97.30	99.40	99.47	99.36	99.25	99.32	99.29
Stubble	98.24	98.13	100.0	100.00	100.00	99.21	100.00	100.0	100.0	99.97	100.0	99.87
Celery	98.97	98.51	99.97	99.78	99.97	99.44	100.00	99.75	100.0	100.0	99.97	99.83
Grapes untrained	98.78	96.98	98.77	84.31	99.99	92.41	99.52	94.79	99.78	93.19	99.88	89.72
Soil vinyard develop	99.32	98.84	99.79	99.79	99.79	99.79	99.81	99.77	99.79	99.77	99.81	99.77
Corn senesced green weeds	99.71	98.44	99.79	99.88	99.66	99.26	99.69	99.17	99.66	99.14	99.69	99.23
Lettuce romaine 4wk	100.00	99.71	100.0	7.99	100.00	89.55	100.00	99.62	100.0	99.81	100.0	95.36
Lettuce romaine 5wk	98.12	98.59	99.95	99.63	99.53	97.70	99.63	99.84	99.53	99.48	99.58	99.37
Lettuce romaine 6wk	98.70	96.60	100.0	99.12	99.89	99.23	100.00	99.01	100.0	99.67	100.0	99.23
Lettuce romaine 7wk	98.81	95.21	99.91	98.87	100.00	96.05	99.91	96.32	99.91	98.59	99.91	99.15
Vinyard untrained	84.02	68.51	97.91	99.65	83.00	96.90	93.75	93.52	89.36	95.03	86.65	97.23
Vinyard vertical trellis	98.10	97.66	98.89	99.50	98.89	99.56	98.89	97.89	98.89	99.05	98.94	98.55
Overall Accuracy (OA)	**97.47**	**96.41**	**99.33**	**94.27**	**98.09**	**97.05**	**98.92**	**97.74**	**98.32**	**97.68**	**98.04**	**97.06**
Average Accuracy (AA)	**97.95**	**96.45**	**99.59**	**91.82**	**98.72**	**97.34**	**99.37**	**98.65**	**99.10**	**98.87**	**98.94**	**98.30**
Computation Time (s)	**854.07**	**849.23**	**968.0**	**952.00**	**882.62**	**825.99**	**864.70**	**821.3**	**817.2**	**799.8**	**933.6**	**903.4**

In summary, the experiments show the effectiveness of pre-processing techniques when applied to the data set, achieving superior performance in classifying hyperspectral images, and one of the best techniques that contributed effectively was the static/dynamic filter technique, which had the highest score in most of the data results.

4. Conclusion

Hyperspectral image classification performance is essential, for applications such as agriculture, geological research, and environmental monitoring, improving. In the past ten years, active learning frameworks have gained a lot of popularity as a way to overcome the difficulties associated with limited labeled samples. CNN+AL has proven to be effective in this particular field among these methods. However, CNN+AL does not preprocess the HSI spectral signatures. In our paper, we proposed using preprocessing techniques, such as different dimensionality reduction techniques and edge-preserving filters, to improve the CNN+AL method. Dimensionality reduction methods help prevent dimensionality and remove redundant and unnecessary information from the data. In this context, dimensionality reduction aids in preventing data duplication, extracting useful features, and reducing resources and training time. While edge-preserving filters aim to reduce noise and smooth the image while preserving the clarity of HSI class boundaries and their information, they also support high classification performances.

We performed different combinations of DR and EPF methods on the three reference datasets used in the methodology: Indian Pines, University of Pavia, and Salinas. Each of these datasets represents a different scenario and set of distinct characteristics. The framework uses five distinct edge-preserving filters while employing PCA and AVG as dimensionality reduction techniques. The proposed method proved that the pre-processing step in active learning is significant and that it outperforms the CNN+AL method in terms of overall and average accuracy. The highest percentage was in Indian Pines and Pavia University when AVG technology was combined with SD Filter, where it gave a result of (99.03%, 98.00%), respectively, while Salinas had the highest result when PCA was combined with WLS, where the result was (99.33%). Although dimensionality reduction methods contribute to reducing the amount of time required to train a CNN, filtering algorithms contribute to increasing the total amount of time required for the CNN+AL framework through their use of computational time.

References

Alcolea, A., Paoletti, M. E., Haut, J. M., Resano, J., & Plaza, A. (2020). Inference in supervised spectral classifiers for on-board hyperspectral imaging: An overview. *Remote Sensing*, *12*(3), 534. https://doi.org/10.3390/rs12030534

Cao, X., Yao, J., Xu, Z., & Meng, D. (2020). Hyperspectral image classification with convolutional neural network and active learning. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(7), 4604–4616.

Chen, M., Wang, Q., & Li, X. (2018). Discriminant analysis with graph learning for hyperspectral image classification. *Remote Sensing*, *10*(6), 836. https://doi.org/10.3390/rs10060836

Chen, Y., Jiang, H., Li, C., Jia, X., & Ghamisi, P. (2016). Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, *54*(10), 6232–6251.

Gupta, V., Sastry, S., & Mitra, S. K. (2020). Hyperspectral image classification using trilateral filter and deep learning. In *2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC)* (pp. 1-6). IEEE.

Ham, B., Cho, M., & Ponce, J. (2015). Robust image filtering using joint static and dynamic guidance. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4823-4831). IEEE.

Haut, J. M., Paoletti, M. E., Plaza, J., Li, J., & Plaza, A. (2018). Active learning with convolutional neural networks for hyperspectral image classification using a new Bayesian approach. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(11), 6440–6461.

He, L., Li, J., Liu, C., & Li, S. (2017). Recent advances on spectral–spatial hyperspectral image classification: An overview and new guidelines. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(3), 1579–1597.

Hong, D., Yokoya, N., Chanussot, J., & Zhu, X. X. (2019). CoSpace: Common subspace learning from hyperspectral-multispectral correspondences. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(7), 4349–4359.

Hong, D., Yokoya, N., Ge, N., Chanussot, J., & Zhu, X. X. (2019). Learnable manifold alignment (LeMA): A semi-supervised cross-modality learning framework for land cover and land use classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 147, 193–205.

Hu, L., Luo, X., & Wei, Y. (2020). Hyperspectral image classification of convolutional neural network combined with valuable samples. Journal of Physics: Conference Series, 1549(5), 052011. https://doi.org/10.1088/1742-6596/1549/5/052011

Hu, Q., Xu, W., Liu, X., Cai, Z., & Cai, J. (2021). Hyperspectral image classification based on bilateral filter with multispatial domain. *IEEE Geoscience and Remote Sensing Letters*, *19*, 1–5. https://doi.org/10.1109/LGRS.2021.3058182

Jia, S., Zhang, X., & Li, Q. (2015). Spectral–Spatial Hyperspectral Image Classification Using $\ell_{1/2}$ Regularized Low-Rank Representation and Sparse Representation-Based Graph Cuts. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *8*(6), 2473–2484.

Joshi, A. J., Porikli, F., & Papanikolopoulos, N. (2009). Multi-class active learning for image classification. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 2372–2379). IEEE.

Kang, X., Li, S., & Benediktsson, J. A. (2013). Spectral–spatial hyperspectral image classification with edge-preserving filtering. *IEEE Transactions on Geoscience and Remote Sensing*, *52*(5), 2666–2677.

Kilik, R. (2021). Histogram-based weighted median filtering used for noise reduction of digital elevation model data. *Acta Geodaetica et Geophysica*, *56*(4), 743–764. https://doi.org/10.1007/s40328-021-00356-2

Kotwal, K., & Chaudhuri, S. (2010). Visualization of hyperspectral images using bilateral filtering. *IEEE Transactions on Geoscience and Remote Sensing*, *48*(5), 2308–2316.

Li, J. (2015). Active learning for hyperspectral image classification with a stacked autoencoders based neural network. In *2015 7th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing* (WHISPERS) (pp. 1–4). IEEE.

Liu, P., Zhang, H., & Eom, K. B. (2016). Active deep learning for classification of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(2), 712–724.

Min, D., Choi, S., Lu, J., Ham, B., Sohn, K., & Do, M. N. (2014). Fast global image smoothing based on weighted least squares. *IEEE Transactions on Image Processing*, *23*(12), 5638–5653.

Santara, A., Mani, K., Hatwar, P., Singh, A., Garg, A., Padia, K., & Mitra, P. (2017). BASS net: Band-adaptive spectral-spatial feature learning neural network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *55*(9), 5293–5301.

Thilagavathi, K., Nagendran, R., & Mary, I. T. B. (2021). Hyperspectral image classification using ensemble average method. In *2021 International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA)* (pp. 1–6). IEEE.

Wang, Q., Chen, M., Zhang, J., Kang, S., & Wang, Y. (2021). Improved active deep learning for semi-supervised classification of hyperspectral image. *Remote Sensing*, *14*(1), 171. https://doi.org/10.3390/rs14010171

Wan, X., & Chen, S. (2023). Hyperspectral image classification using improved multi-scale block local binary pattern and bi-exponential edge-preserving smoother. *European Journal of Remote Sensing*, *56*(1), 2237654. https://doi.org/10.1080/22797254.2023.2237654

Yang, Y., Wu, D., Zeng, L., & Li, Z. (2024). Weighted least square filter via deep unsupervised learning. *Multimedia Tools and Applications*, *83*, 31361–31377. https://doi.org/10.1007/s11042-023-16844-2.

Zhang, D., Kang, J., Xun, L., & Huang, Y. (2019). Hyperspectral image classification using spatial and edge features based on deep learning. *International Journal of Pattern Recognition and Artificial Intelligence*, *33*(09), 1954027. https://doi.org/10.1142/S0218001419540272

Zhu, F., Liang, Z., Jia, X., Zhang, L., & Yu, Y. (2019). A benchmark for edge-preserving image smoothing. *IEEE Transactions on Image Processing*, *28*(7), 3556–3570.