Image processing and machine learning-based classification method for hyperspectral images

Orhan Yaman1 | Hasan Yetiş2 | Mehmet Karaköse2

1 Department of Computer Technology, Firat University, Elazig, Turkey
2 Department of Computer Engineering, Firat University, Elazig, Turkey

Abstract
Hyperspectral images are used to produce effective solutions in many areas due to the number of information they have. However, the number of information they provide may not bring an advantage always, in contrast, it may cause confusion sometimes. In this study, we propose a machine learning-based classification method on the reduced bands using median and mean filters. The proposed method is tested on Indian Pines, Pavia University and Salinas datasets. After applying normalization, median filter and mathematical morphology for each band in an image, a feature matrix with only one band is achieved. Then, the most significant features are selected by using relief feature selection algorithm. The selected features are used for classification with support vector machine and K nearest neighbour. For all methods and datasets, a success rate above 99% in terms of accuracy is achieved. The proposed method has two significant contributions. First, when the proposed method is compared with the similar studies in the literature, it is clearly seen that the features selected by relief algorithm significantly increase the success rate of classification algorithms. Second significant contribution of using relief algorithm is obtaining faster methods by reducing the number of hyperspectral bands.

1 | INTRODUCTION

1.1 | Background

Hyperspectral imaging is a new generation of remote sensing technology that has been developing rapidly in recent years. Hyperspectral technology contains hundreds of bands, unlike multispectral optical detection systems. This imaging method is based on the optical passive remote sensing method. The method of utilizing the properties of the reflection of the rays coming from the sun to the objects on the surface is called passive detection method. It contains wavelengths between 0.4 and 14 \( \mu \text{m} \), which are called the optical region of the electromagnetic spectrum in the optical remote sensing method.

The visible light region is between 0.4 and 0.7 \( \mu \text{m} \), infrared (NIR) is between 0.7 and 1.5 \( \mu \text{m} \), short wavelength infrared is between 1.5 and 3 \( \mu \text{m} \), and medium wavelength infrared is between 3 and 5 \( \mu \text{m} \), and long wavelength infrared is between 5 and 14 \( \mu \text{m} \). In this way, the optical region is divided into five sub-regions in total. Generally, multispectral images contain between 4 and 7 discrete bands and each bandwidth is between 300 and 400 nm. Hyperspectral images consist of hundreds of bands ranging from 10 to 20 nm. While obtaining hyperspectral images, satellites, aircraft equipped with special sensors or portable surface sensors are generally used. In hyperspectral images, a continuous spectrum graph is drawn for each image cell.

1.2 | Motivation

Today, hyperspectral imaging technology is used in a wide range of fields such as geology, medicine, chemistry, agriculture, forestry, space exploration, biomedical, and defence. Recently, the processing of hyperspectral images has been increasing in remote sensing areas. Hyperspectral images in agriculture are used in many fields such as classification of product types, analysis of vegetation, perception of crop and soil conditions, planning of soil structure, assessment of product damages due to environmental factors, and analysis of product quality. Apart

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2021 The Authors. The Journal of Engineering published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology
from agricultural analysis, it is also used in many special areas such as the determination of food quality.

As a result of developing hyperspectral imaging technologies, the studies done in the field of hyperspectral image processing is also increased. The fact that these images contain many bands causes some difficulties in image processing. Recently, deep learning based methods have been developed to overcome these difficulties. However, running deep learning models needs high computing power, which means high cost and energy consumption. Our motivation in this study is to propose a classification method on hyperspectral datasets using computers with low-processing power. When studies in the literature are analyzed, there is no study realized with more than 95% accuracy using SVM and KNN algorithms. In this study, approximately 99% accuracy is achieved by using SVM and KNN algorithms by the help of pre-processing steps.

### 1.3 Literature review

There are many studies in literature on multispectral and hyperspectral images [1–3]. Seniha et al. proposed two methods for extracting the canopy area of the LiDAR sensor from hyperspectral data [4]. In the first method, LiDAR sensor and hyperspectral data are used. Thus, they developed a line of sight algorithm that takes into account the sun's arrival angles and the height of the environment. The second method is to detect shadow areas using only hyperspectral images. They then examined the effects of detected shadows on real data.

Today, image processing methods are developed for agricultural fields and agricultural products using RGB cameras [5–7]. The studies such as diagnosis of agricultural diseases using the Internet of Things and unmanned aerial vehicles [5], detection of weeds on the bird's eye view of agricultural fields [6], measuring the quality of the collected products with image processing techniques [7] are some of them. However, RGB cameras may not sufficient for capturing all useful information of a scene. For this reason, multispectral and hyperspectral cameras are used. Hyperspectral cameras are widely used in agricultural areas as well as in many areas [8,9]. Zang et al. classified hyperspectral images by using original spectral features, extended multi-attribute profiles (EMAPs), and the hybrid of original spectral features and abundance information. In their study, 80.46%, 94.27%, and 88.87% success rates are achieved on the Indian Pines dataset by using original spectral features, EMAPS, and hybrid features respectively [10]. Harikiran et al. used PCA and bidimensional empirical mode decomposition (BEMD) techniques for feature extraction, and deep residual networks (RESNET) for classification. It is reported that PCA-BEMD-RESNET model achieves 99.30% accuracy for Indian Pines dataset [11]. Okwuashi et al. used a method based on deep support vector machine (DSVM) for Indian Pines and University of Pavia datasets. DSVM is compared with some other known classification methods on Indian Pines dataset [12].

Chu et al. proposed a method using three-dimensional convolutional neural networks (3D CNN) with spectral partitioning. It is reported that using spatial spectral features extracted with 3D CNN contributes to the smooth operation of the system with small data. The method that achieves 98.0% classification performance on Indian Pines dataset needs high computational power because it contains many 3D CNN layers [13]. Pathak and Kalita used features derived from spectral and spatial information to classify the hyperspectral images with SVM. The method achieved 91.37% accuracy for Indian Pines dataset [14]. When the methods in the literature are analyzed, it is seen that the studies are generally with low success rate and the studies with high success rate require high computing power. In this study, it is aimed at a method that requires less computing power and higher success rate. The datasets, methods, and success criteria used in the studies in the literature are summarized in Table 1.

### 1.4 Our method

In this study, SVM- and KNN-based methods are proposed using Indian Pines, Pavia, and Salinas datasets. In the proposed method, each band of hyperspectral datasets is converted into a grayscale (0–255) image. To increase classification success, image pre-processing steps are applied to grayscale images. First, the median filter is applied to ensure that the near pixels in the image are more regular. Then, to ensure that the values of the pixels with the same class label converge to each other, mathematical morphology and mean filter are applied. After the pre-processing steps, the 3D hyperspectral matrix is converted to 2D, and the most meaningful features are selected by the help of the Relief algorithm. Finally, the most significant features are used in the classification of segments using SVM and KNN algorithms.

### 1.5 Contributions

The advantages of the proposed method are given below:

- In literature, the success of hyperspectral image classification using KNN and SVM algorithms are lower than the success of CNN algorithms. In this study, better results are obtained in KNN and SVM algorithms by applying the pre-processing steps and Relief algorithm.
- Using the Relief algorithm, only the most meaningful bands are selected instead of using all bands. Thus, the computational complexity of the proposed method is reduced.
- In the literature, deep learning architectures and the computers with high computational power are used for classification of hyperspectral images, while in this study, better results are obtained with light computing methods and a laptop with low computational power.

### 1.6 Study outline

This study continues as follows. In the second part of the study, the datasets used are detailed. The number of spectral bands, the
TABLE 1  Summary of literature studies

| Studies                  | Year | Method    | Dataset                      | Criteria          |
|--------------------------|------|-----------|------------------------------|-------------------|
| Wu et al. [1]            | 2020 | CNN       | Indian Pines, Pavia, Salinas | OA, AA, KAPPA     |
| Wang et al. [2]          | 2018 | LSLRR     | Indian Pines, Pavia, Salinas | OA, AA, KAPPA     |
| Li et al. [2]            | 2018 | CNN       | Indian Pines, Pavia          | OA, AA, KAPPA     |
| Hang et al. [9]          | 2019 | RNN       | Indian Pines, Pavia          | OA, AA, KAPPA     |
| Zhang et al. [10]        | 2018 | SMLR      | Indian Pines                 | OA, AA, KAPPA     |
| Okwuashi et al. [11]     | 2020 | DSVM, SVM, DNN, GMM, KNN, KM | Indian Pines, Pavia | Accuracy          |
| Jonnadula et al. [12]    | 2020 | DRN       | Indian Pines, Pavia          | OA, AA, KAPPA     |
| Chu et al. [13]          | 2019 | 3D-CNN    | Indian Pines, Salinas        | OA, AA            |
| Pathak et al. [14]       | 2019 | SVM       | Indian Pines, Pavia          | OA, AA, KAPPA     |
| Roy et al. [18]          | 2019 | HybridSN  | Indian Pines, Pavia, Salinas | OA, AA, KAPPA     |
| Zhou et al. [22]         | 2019 | Spec-Spat | Indian Pines, Salinas        | Time, OA, AA, KAPPA |
| Duan et al. [23]         | 2019 | MEPOs     | Indian Pines, Pavia          | Pavia             |
| Gong et al. [24]         | 2019 | MS-CNN    | Indian Pines, Pavia, Salinas | Time, OA, AA, KAPPA |
| Chen et al. [25]         | 2019 | CNN       | Indian Pines, Pavia, Salinas | OA, AA, KAPPA     |
| Li et al. [26]           | 2019 | DFFN      | Pavia, Salinas               | OA, AA, KAPPA     |
| Imani et al. [27]        | 2020 | MBFSDA, WJCR-AS | Pavia               | Time, OA, AA, KAPPA |
| Hong et al. [28]         | 2019 | IAP       | Pavia                        | Time, OA, AA, KAPPA |
| Tu et al. [29]           | 2019 | K-SDP     | Pavia, Salinas               | Time, OA, AA, KAPPA |
| Haut et al. [30]         | 2019 | A-ResNet  | Indian Pines, Pavia, Salinas | Time, OA, AA, KAPPA |

way they obtained, and the number of classes for each dataset are explained. In the third part of the study, applied image pre-processing steps, relief feature selection algorithm, and SVM and KNN classification algorithms are clarified. In the fourth section, the results obtained from the proposed method are given and compared with the studies in the literature.

2 | MATERIALS

In this study, three different datasets were used: Indian Pines dataset, Pavia University dataset, and Salinas dataset.

2.1 | Indian Pines dataset

Since Indian Pines dataset is an old dataset, it is used as reference dataset in the literature. This dataset consists of images from the northwest of Indiana state in 1992 [15,16]. Indian Pines dataset was obtained using Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) hyperspectral sensor. This hyperspectral dataset consists of examples with 145 × 145 size and 200 bands. Hyperspectral bands of the Indian Pines dataset have a spectral range of 0.4–2.5 µm. Since the capturing distance of the Indian Pines dataset is about 20 meters, it is a low-resolution dataset with 16 vegetation classes. Artificial RGB image, reference image and labelled data types of Indian Pines dataset are given in Figure 1.

2.2 | Pavia University dataset

This dataset was obtained by capturing hyperspectral images of the University of Pavia in Pavia, Italy. Pavia University data set was obtained using the ROSIS-03 (Reflective Optics System Image Spectrometer) hyperspectral sensor. This hyperspectral
2.3 Salinas dataset

The Salinas data set was created by taking the hyperspectral images of the Salinas Valley in California. This data set was taken with the help of AVIRIS hyperspectral sensor. This hyperspectral dataset consists of examples with 512 × 217 size and 204 bands [17,18]. Hyperspectral bands of the Pavia University dataset have a spectral range of 430 to 860 nm. Since the capturing distance of the dataset is about 3.7 meters, it is a high-resolution dataset with 16 different labelled classes. Artificial RGB image, reference image and labelled data types of Salinas data set are given in Figure 3.

3 THE PROPOSED METHOD

In this study, a method providing high classification success for three different hyperspectral datasets is proposed. The method generally consists of obtaining image from the hyperspectral dataset, image pre-processing, feature extraction, feature selection and classification steps. The block diagram of the proposed method is given in Figure 4.

3.1 Image obtaining from hyperspectral dataset

Hyperspectral images contain wavelengths between 0.4 and 14 µm, called the optical region of the electromagnetic spectrum. Among these wavelengths, hyperspectral datasets were created by taking remote sensing data from many different bands. First, each band in the dataset is normalized between 0 and 255 using (1), where \( X \) is the current pixel value, \( X_{\text{min}} \) is the smallest pixel value in the current band, and \( X_{\text{max}} \) is the largest pixel value in the current band:

\[
X' = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \times 255. \tag{1}
\]

The normalization process is applied for each band of each image in the dataset. This process results in obtaining a new dataset that has values in the range of 0–255.

3.2 Image pre-processing

After converting the hyperspectral images to grayscale images, image pre-processing steps are applied. First of all, median filter, which is widely used for noise removal, is used on the grayscale images to obtain more regular distributed images. The median filter is a non-linear filter type, and it helps to remove the disadvantage of small pixel value changes in a neighbourhood besides removing noise. This paper uses it for making equal the pixels with close values. To apply median filter, a mask with 3 × 3 window size is shifted on the image. The pixel values covered by this mask are sorted and the middle value is assigned to the center pixel. In Figure 5, the application of the median filter with 3 × 3 window size is given.
The process visualized in Figure 5 is applied to each grayscale band image obtained from the hyperspectral dataset. Then, one of the most important applications of mathematical morphology, dilation, is applied on the image. Dilation process is a morphological process that enlarges or thickens the object [19–21]. The purpose of the dilation process is to fill the holes and gaps formed on the image due to the noises and soften the corner points. In this study, a $3 \times 3$ dimension structural element with all pixels are 1 is used. After the dilation process, mean filter is applied on the image. The aim of applying mean filter is to reduce the amount of change between one pixel and the other pixels on the image. In mean filter, first, the window size is determined. After, the average of the values in the scope of the window is calculated. Finally, the calculated mean value is our new centre value. For this convolutional method, we choose a mask with $5 \times 5$ window size. In Figure 6, the mean filter application is shown with a $5 \times 5$ window size.

### 3.3 Feature extraction

In this study, grayscale images between 0 and 255 for each band in hyperspectral datasets. Each data in dataset consists of many bands. For feature extraction from these multiple bands, all bands should be combined into a single feature matrix. Consider that a data with $K$ bands, and each band has $N \times M$ size. To achieve feature matrix, the columns in a band are aligned consecutively. Consequently, the image matrix is transformed into a column vector with length $N \times M$. After applying the process for all bands, $K$ column vectors, each has $N \times M$ length, are obtained. By ordering these vectors horizontally, the feature matrix is obtained ($N \times M \times K$). This process is shown in Figure 7.

As shown in Figure 7, 3D hyperspectral images consisting of multiple bands are transformed into 2D feature matrix. For example, for a data with $145 \times 145$ size and 200 bands in Indian Pines, a feature matrix with $21025 \times 200$ size is created. Similarly, for each of the data in the Pavia University dataset whose size is $610 \times 340 \times 103$, a feature matrix with dimension $207,400 \times 103$; for each of the data in the Salinas dataset
whose size is $512 \times 217 \times 204$, a feature matrix with dimension $111,104 \times 204$ are created. In addition, extra one column for expressing the class label is added.

### 3.4 Feature selection

In this study, relief algorithm is used to reach the most significant features [15]. The relief feature selection algorithm, commonly used in the literature, was developed by Kira et al. According to this algorithm, the weight value is initially set to 0 for each feature. Weights of these features are updated by selecting random data at each step. Thus, the most weighted features are determined as most significant features at the end of the iterations.

### 3.5 Classification

After determination of most significant features, SVM and KNN algorithms are used to classify the samples in the datasets. SVM and KNN algorithms are frequently preferred successful algorithms in classification algorithms. The SVM algorithm is used to examine the connection between variables. It aims to determine the distinction between points by creating an $N$-dimensional space. This paper prefers SVM Cubic algorithm. The KNN algorithm makes classification by looking at the class of its neighbour, whose Euclidean distance is closest. We prefer KNN fine version which is an implementation of KNN algorithm [16]. Generally, in this study, it is done classification with SVM cubic and KNN fine algorithms using MATLAB classification learner application. The parameters used in the classifiers are given in Table 2.

### 3.6 Performance analysis

By running the classification algorithms 100 times with the settings given in Table 2, accuracy, precision, recall, geometric mean, and F-measure values are calculated. The equations for calculating the metrics are given by (2)–(6), where TP means the correct predictions of positive classed data, TN means the correct predictions of negative classed data, FP means the false predictions of positive classed data, and FN means the false predictions of negative classed data:

\[
\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (2)
\]

\[
\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (3)
\]

\[
\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (4)
\]

\[
\text{Geometric mean} = \sqrt{\frac{\text{TP} \times \text{TN}}{(\text{TP} + \text{FN}) \times (\text{TN} + \text{FP})}} \quad (5)
\]

\[
F\text{-measure} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \quad (6)
\]

### 4 EXPERIMENTAL RESULTS

In this study, hyperspectral image classification method is proposed and the proposed method is tested on Indian Pines, Pavia University and Salinas datasets. MATLAB classification learner toolbox is used and the results are obtained by 10-fold cross validation. The result images belong to steps such as obtaining grayscale images, applying median filter, applying dilation, and applying mean filter are given by Figures 8–10. It can be seen that sequential use of the median filter, dilation, and mean filter make the images segmented.

As a result of the operations shown in Figures 8–10, the 3D hyperspectral dataset is converted into a 2D feature matrix. The most significant features are selected from the feature matrix by using the relief algorithm. After, classification is made using SVM and KNN algorithms. The classification performances of the proposed method on Indian, Pavia, and Salinas datasets are given in Table 3.

The results given in Table 3 were obtained after running 100 iterations. When the classification results for the Indian data set are examined, we can see that the best value for SVM Cubic is 99.69%, and the best value for KNN Fine is 99.76%. On the
| Band 1  | Grayscale image | Median Filter (3x3) | Dilation Process | Mean Filter (5x5) |
|--------|-----------------|---------------------|-----------------|-------------------|
|        | ![Grayscale Image](image1) | ![Median Filter](image2) | ![Dilation Process](image3) | ![Mean Filter](image4) |
| Band 200 | ![Grayscale Image](image5) | ![Median Filter](image6) | ![Dilation Process](image7) | ![Mean Filter](image8) |

**Figure 8** Applying median filter, dilation, and mean filter on grayscale image obtained from first and last band of a sample in Indian Pines dataset

| Band 1 | Grayscale image | Median Filter (3x3) | Dilation Process | Mean Filter (5x5) |
|--------|-----------------|---------------------|-----------------|-------------------|
| ![Grayscale Image](image9) | ![Median Filter](image10) | ![Dilation Process](image11) | ![Mean Filter](image12) |
| Band 103 | ![Grayscale Image](image13) | ![Median Filter](image14) | ![Dilation Process](image15) | ![Mean Filter](image16) |

**Figure 9** Applying median filter, dilation, and mean filter on grayscale image obtained from first and last band of a sample in Pavia University dataset
Pavia dataset, it is achieved 99.40% success for SVM cubic and 99.04% success for KNN fine. For Salinas dataset, SVM cubic and KNN fine classification successes are calculated as 99.96% and 99.95%, respectively.

5 | DISCUSSION

In this study, Hyperspectral image classification is made using Indian Pines, Pavia, and Salinas datasets. Image processing is performed on multi-band hyperspectral images. After image processing steps, using the relief algorithm, the most significant bands for classification are selected among the hyperspectral bands. For Indian Pines dataset, only 65 hyperspectral bands selected by relief algorithm among 200 hyperspectral bands in total are used. For the Pavia dataset with 103 hyperspectral bands, the most significant 40 hyperspectral bands selected by the relief algorithm are used. Similarly, for Salinas dataset with 204 hyperspectral bands, only 55 most significant hyperspectral bands are selected.

With selecting the most significant bands, unnecessary bands are eliminated and a performance increase is achieved in case of time. Furthermore, the classification algorithms give better results without useless bands for classification. This paper uses SVM cubic and KNN fine algorithms. The performance results of the proposed method and the results of the studies in the literature are given in Table 4.

When the overall accuracy (OA), average accuracy (AA), and Kappa coefficients (K) parameters given in Table 4 are examined, it is seen that the proposed method gives successful results. Its performance in terms of success and speed are better understood when compared with the current method.
### TABLE 3
The performance metrics of the proposed method on Indian, Pavia, and Salinas dataset

| Dataset       | Class        | Accuracy  | Precision | Recall  | Geometric mean | F score  |
|---------------|--------------|-----------|-----------|---------|----------------|----------|
| Indian Pines  | SVM cubic    | Best      | 99.6925   | 99.7485 | 98.7976        | 98.7840  | 99.2605  |
|               |              | Mean      | 99.6406   | 99.3788 | 98.7102        | 98.6954  | 99.0432  |
|               |              | Std       | 0.0241    | 0.2370  | 0.0632         | 0.0648   | 0.1309   |
| Pavia         | KNN fine     | Best      | 99.7619   | 99.7591 | 99.7799        | 99.7793  | 99.7591  |
|               |              | Mean      | 99.6866   | 99.6575 | 99.6203        | 99.6179  | 99.6388  |
|               |              | Std       | 0.0290    | 0.1091  | 0.1165         | 0.1187   | 0.0810   |
| Salinas       | SVM cubic    | Best      | 99.4055   | 99.3611 | 98.6780        | 98.6868  | 99.0122  |
|               |              | Mean      | 99.3602   | 99.3190 | 98.6149        | 98.6046  | 98.9657  |
|               |              | Std       | 0.0213    | 0.0343  | 0.0439         | 0.0449   | 0.0341   |
| Pavia         | KNN fine     | Best      | 98.9922   | 98.6647 | 97.8789        | 97.8589  | 98.2702  |
|               |              | Mean      | 99.0421   | 98.7662 | 97.9784        | 97.9600  | 98.3516  |
|               |              | Std       | 0.0199    | 0.0353  | 0.0450         | 0.0461   | 0.0362   |
| Salinas       | SVM cubic    | Best      | 99.9471   | 99.9436 | 99.9299        | 99.9298  | 99.9367  |
|               |              | Mean      | 99.9588   | 99.9700 | 99.9614        | 99.9613  | 99.9653  |
|               |              | Std       | 0.0078    | 0.0105  | 0.0123         | 0.0123   | 0.0109   |
|                |              | Mean      | 99.9467   | 99.9591 | 99.9488        | 99.9487  | 99.9539  |
|                |              | Std       | 0.0051    | 0.0063  | 0.0068         | 0.0069   | 0.0062   |

### TABLE 4
Performance comparison of the proposed method and literature studies

| Reference and year | Method   | Feature                                      | Dataset      | Time (minute) | OA (%)  | AA (%)  | KAPPA (%) |
|--------------------|----------|----------------------------------------------|--------------|---------------|---------|---------|------------|
| Zhou et al. [22]   | Spec-Spat| MATLAB, 3.5 GHZ 8 core, 64GB RAM             | Indian Pines | 150           | 95.81   | 97.38   | 95.30      |
|                    |          |                                              | Pavia        | 108           | 97.59   | 97.66   | 96.86      |
|                    |          |                                              | Salinas      | 210           | 96.07   | 97.56   | 96.78      |
| Hang et al. [9]    | RNNs     | –                                            | Indian Pines | –             | 91.79   | 95.94   | 90.62      |
|                    |          |                                              | Pavia        | –             | 90.30   | 87.97   | 86.26      |
| Duan et al. [23]   | MEPOs    | MATLAB, 2.6 GHz, 8GB RAM                      | Pavia        | –             | 87.82   | 87.07   | 84.30      |
| Gong et al. [24]   | MS-CNN   | 3.6 GHz, Intel i7, 64 GB RAM                  | Indian Pines | 13.28         | 99.08   | 99.52   | 98.83      |
|                    |          |                                              | Pavia        | 11.33         | 99.46   | 99.39   | 99.27      |
|                    |          |                                              | Salinas      | 10.70         | 97.51   | 98.85   | 97.88      |
| Chen et al. [25]   | CNN      | Python, 2.5 GHz, i5, 8 GB RAM                 | Indian Pines | –             | 89.01   | 72.88   | 87.16      |
|                    |          |                                              | Pavia        | –             | 93.88   | 88.19   | 92.15      |
|                    |          |                                              | Salinas      | –             | 94.65   | 93.09   | 93.87      |
| Li et al. [26]     | DFFN     | –                                            | Pavia        | –             | 98.57   | 99.16   | 98.08      |
|                    |          |                                              | Salinas      | –             | 99.71   | 99.78   | 99.67      |
| Imani et al. [27]  | MBFSDA   | –                                            | Pavia        | 0.45          | 91.39   | 93.15   | 88.78      |
|                    | WJCR-AS  | –                                            | Pavia        | 26.3          | 94.47   | 97.83   | 92.82      |

(Continues)
As can be seen in the literature studies, the accuracy rates of SVM and KNN algorithms are low for the mentioned datasets. In this study, we aimed to increase the performance of SVM and KNN by eliminating unnecessary bands. In this way, the results are obtained with high accuracy in a practical, and faster way. It is seen that the accuracies of the SVM and KNN methods, which is around 80% before, are increased to 99% with the study. The proposed method focuses on studies that increase the success of the current methods rather than proposing a completely new method. The used pre-processing steps and relief algorithm are tested also for decision tree and Bayesian classifier, but the highest accuracies are obtained for KNN and SVM. Moreover, the highest accuracy is achieved with cubic SVM in SVM algorithms and fine KNN in KNN algorithms.

The accuracies and running times of the methods are calculated for both cases, which are with image pre-processing steps and without image pre-processing steps. The obtained results for different datasets are given in Table 5.

When image pre-processing steps are not applied, both SVM and KNN classifiers give results with lower accuracy. Further-

### TABLE 4 (Continued)

| Reference and year | Method | Feature | Dataset | Time (minute) | OA (%) | AA (%) | KAPPA (%) |
|--------------------|--------|---------|---------|---------------|--------|--------|-----------|
| Hong et al. [28]   | IAP    | MATLAB, i7, 2.6 GHz, 16 GB RAM | Pavia   | 11.2          | 95.78  | 93.15  | 94.41     |
| Tu et al. [29]     | K-SDP  | MATLAB, 2.4 GHz, 8 GB RAM      | Pavia   | 30.72         | 79.55  | 79.50  | 75.42     |
| Haut et al. [30]   | A-ResNet | Pytorch, Intel i7, 4.2 GHz, 40 GB RAM | Indian Pines | 1.54   | 98.75  | 97.05  | 98.58     |
|                    |        |         | Salinas | 35.22         | 83.94  | 90.80  | 83.89     |
| Okwuashi et al. [11]| SVM    | –       | Indian Pines | 3.43   | 99.86  | 99.76  | 99.82     |
|                    |        |         | Salinas | 4.79          | 99.85  | 99.83  | 99.83     |
|                    | KNN    | –       | Indian Pines | 76.03 | 73.52  | –      | –         |
|                    |        |         | Salinas | 4.79          | 99.85  | 99.83  | 99.83     |
| The proposed method | SVM Cubic | MATLAB, i5-4300 M, 2.6 GHz, 8 GB RAM, Windows 10 | Indian Pines | 4.13  | 99.71  | 99.64  | 99.69     |
|                    |        |         | Pavia   | 20.98         | 99.38  | 99.36  | 99.40     |
|                    |        |         | Salinas | 22.26         | 99.93  | 99.94  | 99.96     |
|                    | KNN fine| –       | Indian Pines | 3.31  | 99.70  | 99.68  | 99.76     |
|                    |        |         | Pavia   | 16.81         | 90.02  | 98.99  | 99.04     |
|                    |        |         | Salinas | 17.84         | 99.93  | 99.94  | 99.95     |

### TABLE 5

| Dataset name | Method | Accuracy (%) | Time (minute) |
|--------------|--------|--------------|---------------|
| Indian Pines | Cubic SVM with preprocessing | 99.6 | 4.13 |
|              | Fine KNN with preprocessing | 99.7 | 3.31 |
|              | Cubic SVM without preprocessing | 91.2 | 5.68 |
|              | Fine KNN without preprocessing | 82.9 | 4.55 |
| Pavia        | Cubic SVM with preprocessing | 99.4 | 20.98 |
|              | Fine KNN with preprocessing | 99.0 | 16.81 |
|              | Cubic SVM without preprocessing | 95.1 | 25.67 |
|              | Fine KNN without preprocessing | 92.3 | 20.54 |
| Salinas      | Cubic SVM with preprocessing | 99.9 | 22.26 |
|              | Fine KNN with preprocessing | 99.9 | 17.84 |
|              | Cubic SVM without preprocessing | 94.4 | 27.92 |
|              | Fine KNN without preprocessing | 91 | 22.38 |
more, when the pre-processing steps are not applied, the relief algorithm chooses more features, and the classification process needs more time.

The advantages of the proposed hyperspectral image classification algorithm are given below:

- Pre-image processing before classification in the proposed method increases the success of the method.
- The most significant bands are selected using the relief algorithm on the hyperspectral dataset which has tens of bands. Thus, the proposed method is provided to work faster.
- The proposed method requires less resources for classification by using the most significant bands with feature selection instead of all hyperspectral bands.

6 | CONCLUSIONS

In this study, a method based on image processing and machine learning is proposed for hyperspectral image classification, which is important for the analysis of agricultural areas. According to the proposed method, first, the hyperspectral images which have multiple bands are converted into grayscale images. Then image pre-processing methods are applied on grayscale images. Later, features are extracted and the most significant features are selected by using the relief algorithm. Selected features are used in classification algorithms. In this study, two different classification methods which are SVM and KNN are employed. SVM cubic and KNN fine versions of these algorithms are chosen. The methods are tested on three different agricultural hyperspectral datasets which are Indian Pines, Pavia, and Salinas. For three different datasets, the best SVM cubic classification result is 99.96%, and the best KNN fine result is 99.95%. When the studies in the literature are examined, it is clearly seen that the image pre-processing and feature selection steps help to improve the success rate.

ACKNOWLEDGEMENTS

This work was supported by Scientific Research Projects Coordination Unit of Firat University (Project number: MF:19.31).

ORCID

Orhan Yaman  https://orcid.org/0000-0001-9623-2284
Mehmet Karaköse  https://orcid.org/0000-0002-3276-3788

REFERENCES

1. Wu, Y., et al: Semi-Supervised hyperspectral image classification via spatial-regulated self-training. Remote Sens. 12(1), 159 (2020)
2. Wang, Q., Xiang, H., Xuelong, L.: Locality and structure regularized low rank representation for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57(2), 911–923 (2019)
3. Paolletti, M.E., et al: Deep learning classifiers for hyperspectral imaging: A review. ISPRS J. Photogramm. Remote Sens. 158, 279–317 (2019).
4. Yüksel, S.E., Boyaci, M.: LiDAR sensörüün hiperspektral veri veren gölgeli alan elde etme avantajı etkisi. Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi 24(2), 198–204 (2018)
5. Gao, D., et al: A framework for agricultural pest and disease monitoring based on Internet-of-Things and unmanned aerial vehicles. Sensors 20(5), 1487 (2020)
6. Gai, J., Lie, T., Brian, L.S.: Automated crop plant detection based on the fusion of color and depth images for robotic weed control. J. Field Rob. (2019)
7. Abbas, H.M.T., et al: Automated sorting and grading of agricultural products based on image processing. In: IEEE 8th International Conference on Information and Communication Technologies (ICICT), (2019)
8. Mohan, A., Venkatesan, M.: Spatial data-based prediction models for crop yield analysis: A systematic review. In: Emerging Research in Data Engineering Systems and Computer Communications, Singapore, 341–352 (2020)
9. Hang, R., et al: Cascaded recurrent neural networks for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57(8), 5384–5394 (2019)
10. Zhang, X., Wenchao, Q.: Hyperspectral image classification based on extended morphological attribute profiles and abundance information. In: IEEE 9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), (2018)
11. Oluwaziki, O., Christopher, E.N.: Deep support vector machine for hyperspectral image classification. Pattern Recognit. 103, 107298, (2020)
12. Jonnadula, H., et al: Hyperspectral image classification bi-dimensional empirical mode decomposition and deep residual networks. In: IEEE International Conference on Artificial Intelligence and Signal Processing (AISP), (2020)
13. Chu, R.S.W., et al: Convolution based spectral partitioning architecture for hyperspectral image classification. In: IEEE International Geoscience and Remote Sensing Symposium (IGARSS) (2019)
14. Pathak, D.K., Sanjib, K.K.: Spectral Spatial Feature Based Classification of Hyperspectral Image Using Support Vector Machine. IEEE 6th International Conference on Signal Processing and Integrated Networks (SPIN), (2019)
15. Kira, K., Rendell, L.A.: The feature selection problem: Traditional methods and a new algorithm. In: Proceedings of the 10th National Conference on Artificial Intelligence 2, 129–134 (1992)
16. Yaman, O., Ertäm, F., Tuner, T: Automated Parkinson’s disease recognition based on statistical pooling method using acoustic features. Med. Hypotheses. 109483, 135, (2020)
17. Tarabalka, Y., Chanussot, J., Benediktsson, J.A.: Segmentation and classification of hyperspectral images using watershed transformation. Pattern Recognit. (2010), 43, 2367–2379
18. Roy, S.K., et al: Hybrid: Exploiting 3-d-2-d cnn feature hierarchy for hyperspectral image classification. IEEE Geosci. Remote Sens. Lett. 17(2), 277–281 (2019)
19. Yaman, O., Karakose, M.: Development of image processing based methods using augmented reality in higher education. In: IEEE 15th International Conference on Information Technology Based Higher Education and Training (ITHET), pp. 1–5, IEEE, Piscataway, NJ (2016)
20. Tastimur, C., et al: Rail defect detection and classification with real time image processing technique. Int. J. Comput. Sci. Softw. Eng. 5(12), 283 (2016)
21. Karakose, M., et al: A new computer vision based method for rail track detection and fault diagnosis in railways. Int. J. Mech. Eng. Robot. Res. 6(1), 22–27 (2017)
22. Zhou, P., et al: Learning compact and discriminative stacked autoencoder for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57(2), 4823–4833 (2019)
23. Duan, P., et al: Fusion of multiple edge-preserving edge-preserving operations for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57(12), 10336–10349 (2019)
24. Gong, Z., et al: A CNN with multiscale convolution and diversified metric for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57(6), 3599–3618 (2019)
25. Shen, Y., et al: Automatic design of convolutional neural network for hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57(9), 7048–7066 (2019)
26. Li, S., et al: Deep learning for hyperspectral image classification: An overview. IEEE Trans. Geosci. Remote Sens. 57(9), 6690–6709 (2019)
27. Imani, M., Ghassemian, H.: An overview on spectral and spatial information fusion for hyperspectral image classification: Current trends and challenges. Inf. Fusion 59, 59–83 (2020)
28. Hong, D., et al.: Invariant attribute profiles: A spatial-frequency joint feature extractor for hyperspectral image classification. arXiv:1912.08847, (2019)

29. Tu, B., et al.: Spatial density peak clustering for hyperspectral image classification with noisy labels. IEEE Trans. Geosci. Remote Sens. 57(7), 5085–5097 (2019)

30. Haut, J.M., et al.: Visual attention-driven hyperspectral image classification. IEEE Trans. Geosci. Remote Sens. 57(10), 8065–8080 (2019)