

Hyperspectral Image Band Selection Based on CNN Embedded GA (CNNeGA)

Mohammad Esmaeili, Dariush Abbasi-Moghadam , Alireza Sharifi , Aqil Tariq , and Qingting Li 

Abstract—Hyperspectral images (HSIs) are a powerful source of reliable data in various remote sensing applications. But due to the large number of bands, HSI has information redundancy, and methods are often used to reduce the number of spectral bands. Band selection (BS) is used as a preprocessing solution to reduce data volume, increase processing speed, and improve methodology accuracy. However, most conventional BS approaches are unable to fully explain the interaction between spectral bands and evaluate the representation and redundancy of the selected band subset. This study first examines a supervised BS method that allows the selection of the required number of bands. A deep network with 3D-convolutional layers embedded in a genetic algorithm (GA). The GA uses embedded 3D-CNN (CNNeGA) as a fitness function. GA also considers the parent check box. The parent check box (parent subbands) is designed to make genetic operators more effective. In addition, the effectiveness of increasing the attention layer to a 3D-CNN and converting this model to spike neural networks has been investigated in terms of accuracy and complexity over time. The evaluation of the proposed method and the obtained results are satisfactory. The accuracy improved from 6% to 21%. Accuracy between 90% and 99% has been obtained in each evaluation mode.

Index Terms—Attention layer, band selection (BS), convolution neural networks (CNNs), embedded algorithm, genetic algorithm (GA), hyperspectral image (HSI), spiking neural networks (SNNs).

I. INTRODUCTION

REMOTE sensing can be used to gather information about things or events that are happening on earth's surface [1]. Hyperspectral image (HSI) is a method that employs spectral differences to separate terrestrial objects as a result of advancements in image spectrometer technology. However, HSI

Manuscript received 17 November 2022; revised 3 January 2023; accepted 31 January 2023. Date of publication 6 February 2023; date of current version 16 February 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 42071321. (Corresponding authors: Aqil Tariq; Qingting Li.)

Mohammad Esmaeili and Dariush Abbasi-Moghadam are with the Department of Electrical Engineering, Shahid Bahonar University of Kerman, Kerman 76169-14111, Iran (e-mail: m-esmaeili@eng.uk.ac.ir; abbasimoghadam@uk.ac.ir).

Alireza Sharifi is with the Department of Surveying Engineering, Faculty of Civil Engineering, Shahid Rajaei Teacher Training University, Tehran 1678815811, Iran (e-mail: a_sharifi@sru.ac.ir).

Aqil Tariq is with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China, and also with the Department of Wildlife, Fisheries and Aquaculture, Mississippi State University, Mississippi State, MS 39762 USA (e-mail: aqiltariq@whu.edu.cn; at2139@msstate.edu).

Qingting Li is with the Airborne Remote Sensing Center, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China (e-mail: liqt@radi.ac.cn).

Digital Object Identifier 10.1109/JSTARS.2023.3242310

creates an enormous amount of data that can be challenging to process fast. Therefore, it can be difficult to apply standard image processing methods designed for multispectral images [2], [3]. Without a question, HSI data offer scientists and researchers a wealth of knowledge. However, because it takes a lot of computer power to analyze these data, it is frequently difficult to make the most of HSI's potential. Large datasets maybe common image processing algorithms inefficient and classification accuracy may decrease. Several important factors can hinder classification in large datasets, including the following.

- 1) A class imbalance in a large dataset.
- 2) The existence of noise in educational data.
- 3) Variety and a high number of classes.
- 4) Nonintegrated category (nonintegrated pixels in each class).
- 5) Closeness and similarity of classes, etc.

New dimensional reduction techniques and procedures are required to process HSI properly to get beyond these obstacles. Dimension reduction is the technique that minimizes the number of dimensions in the data while keeping most of the information. This will help open the full potential of HSI data while minimizing costs and time requirements [4], [5], [6]. One can reduce HSI dimensions by choosing bands or extracting features [7]. By applying specified criteria, feature extraction transforms original data into a different attribute space [8], [9]. Most feature extraction approaches combine all of the major bands linearly. Inappropriate criterion adoption can be troublesome since it makes it more challenging to evaluate the findings [10]. Many feature extraction techniques include the preprocessing phase of band selection (BS). Choosing a subset of desired bands from the main bands can be used to enhance the results' interpretability. This keeps the spectral meaning intact and boosts efficiency [11]. Feature extraction methods are essential for reducing the amount of data needed to be processed. It enables us to deal with less data while yet maintaining the most crucial information. However, it should be noted that not all feature extraction methods are created equal. It is important to note that these methods can often compromise or distort vital features in the data. This can significantly impact how these data are interpreted physically [12], [13].

When working with high-dimensional data, such as that included in HSI, feature selection is crucial. We may simplify the data and make it simpler to comprehend and deal with by lowering the number of dimensions. Feature selection techniques assist maintain the data's physical interpretation while lowering dimension. This is an essential step in HSI analysis and should

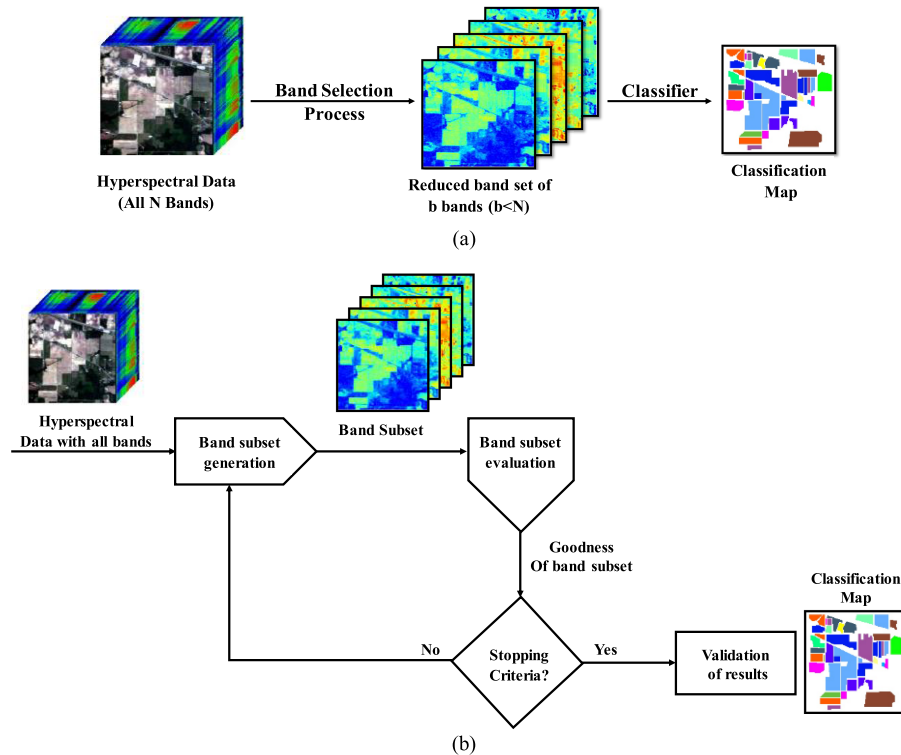


Fig. 1. Comprehensive guide to the BS process. (a) General framework of the BS process. (b) Stages of the BS process.

be given careful consideration [14]. By doing this, we can ensure that we are selecting features that are related to the original data and preserving its meaning. Because the HSI is not useful or significant based on the ground cover classes, irrelevant and extra bands are frequently eliminated when choosing bands for HSI. However, analyzing these data can be rather difficult owing to the Hughes effect when the number of cases is significantly smaller than the number of characteristics, as is frequently the case with HSI data. As illustrated in Fig. 1, [(a) BS of the basic framework and (b) stages of the BS process], the general framework and process for BS typically consist of four major processes. The steps are as follows:

- 1) production of band subsets;
- 2) evaluation of band subsets;
- 3) stop criteria; and
- 4) validation of results [15].

HSI bands are an important part of the identification process for many people. It is crucial to choose carefully when choosing a band because doing so can be essential to success. Following these steps carefully and thoughtfully ensures that your final selection for HSI bands will be both relevant and accurate [16], [17]. Two main types of BS methods are supervised and unsupervised. Supervised methods require prior knowledge of the data, whereas unsupervised methods do not. The main difference between these two types of methods is how the criteria for BS are constructed. Supervised methods use predetermined design criteria, whereas unsupervised methods use data itself to determine which bands are most important. The best way for your application must be chosen because each sort of method

has benefits and drawbacks [18]. The proposed models are more accurate when a smaller number of bands are processed. The results show that the number of spectral bands selected, the errors obtained, and the trend in the proposed models represent local or global minimums that determine the performance and accuracy of the optimal model. In addition, the problem of choosing the monitored method is considered by choosing the method that constitutes the most accurate model [19].

II. RELATED WORK

Principal component analysis (PCA) and spectral eigen decomposition are potent methods for dimensionality reduction and data analysis. PCA is a technique that finds the linear combination of variables that best explained the variance in a dataset. This can be used to reduce the number of dimensions in a dataset while preserving most of the information. In [20], PCA is used in the spectral dataset and is used to prioritize bands to maximize variance or SNR. In [21], combining supervised and unsupervised methods has various benefits to create a precise HSI BS system. Instance-based supervised neural computing was integrated with unsupervised techniques, such as information entropy, information divergence, and PCA-based band ranking. This produced superior performance results than using only supervised or unsupervised techniques. Combining several methods enables a more precise selection of bands most relevant to the task at hand, improving classification accuracy. In [22], four linear constraints were utilized to minimize band correlation: spectral angle, normalized mutual information (MI),

Pearson's correlation coefficient, and Spearman's rank correlation coefficient. In [23], an unmixing-analysis-based alternative separability index approach was used for HSI BS. With this approach, correlated bands that cannot be effectively separated are forced to be removed from the analysis. Results in a more accurate and reliable final image. In [24], the interband redundancy was eliminated using methods of band decorrelation based on divergence. The first step was calculating the divergence between each pair of channels. The channels with lower divergence values were then selected for each band's representative channels. This approach resulted in improved image quality and a reduction in processing time. Finally, the remaining bands were prioritized according to their significance. The authors utilized two approaches; a clustering-based method and a ranking-based method to choose which bands to use for HSI. It was improved to make the "rapid density-peak-based clustering" approach more appropriate for HSI BS. The "isolated-point-stopping criterion" determines the number of bands that are chosen. When used on two independent datasets, this methodology surpassed earlier techniques, according to this study [25].

In [26], the pointwise-ranking-based HSI BS proposed by these authors is very efficient and can be used to select an appropriate subset of bands for further analysis or classification tasks. This approach, based on the average correlation of labels supplied by a trained nonhomogeneous hidden Markov chain model performed to wavelet processed HSI data, ranks spectral bands. The importance of each band in an HSI is evaluated in [27] and [28] in accordance with MI between the HSI bands and a common reference band. One of the most efficient ways is the clustering-based steering-based HSI band reduction method [10]. This method chooses representative bands from the clusters based on information metrics, such as MI or Kullback–Leibler divergence. This lowers the number of dimensions while enabling a more accurate depiction of the data. A brand-new unsupervised band reduction method called BandClust has been studied in [29]. This approach splits each band interval into two disjoint contiguous subbands based on minimization criteria of MI between averaged subbands. As a result, there are fewer bands while the majority of the spectral data is still preserved. One such semisupervised technique, band clustering, which uses class spectral signatures for band clustering, was presented by the authors in [30]. Each cluster center is used as a representative band once outliers from the generated clusters have been eliminated. A system for automatically removing water-absorbed, low discriminating, and high SNR bands was put forth [31]. The authors described feature mining as a useful method for identifying certain representational bands [32]. It accomplishes this by examining the connection between each band cluster. According to Zeng et al. [33], convolutional autoencoders are employed to learn the features of each data point in the cluster. The ideal band is chosen for each cluster using this information. In [34], the HSI BS issue was addressed by the authors using graph algorithms. Data analysis may be done mathematically using the rough set theory. Liu et al. [35] used to identify the most important bands in an HSI. This is done by first reducing the number of dimensions in the HSI so that it can be more easily analyzed. The significance of each band is then determined

using a forward greedy search technique. The algorithm starts with the most significant band and searches for the next most significant band until all of the bands have been evaluated. In [36], the performance of the wrapper-based semisupervised HSI BS techniques was improved by the authors through the introduction of the usage of guided filter pseudolabels. This approach is based on a novel two-stage filtering procedure that first uses unsupervised learning algorithms to automatically identify relevant bands for further analysis and then uses a supervised classifier to label them. In [37], a three-step strategy is proposed. This approach begins with breaking down HSI bands into band subgroups. This can be done by dividing the spectrum into regions based on wavelength and then assigning each pixel to a band according to its location in the spectrum. The second step is to select nonredundant bands from the decomposed subsets. This can be done by ranking the bands based on their spectral information content and selecting only the top N bands, where N is a user-defined parameter. The third step is to use these selected bands for classification or other analysis tasks. Evolutionary methods are a family of search algorithms that use principles of natural selection and genetics to optimize solutions. Numerous evolutionary techniques have been used for effective band searching, including the genetic algorithm (GA) [38], particle swarm optimization [39], and firefly algorithm [40]. An innovative-supervised-filter-based method for BS using neural networks is suggested in [41]. A binary single-layer neural network classifier creates a classification between each class in the dataset and the rest of the data for each class in the dataset. The procedure of choosing the bands is then class-oriented since the largest and lowest weight bands are chosen next. Up until the predetermined number of bands is reached, this procedure iterates.

In [42], semisupervised BS using an upgraded Levy flight based variant of the GA is performed. In the suggested semisupervised strategy, both spectral similarity and spatial proximity are used to increase the number of training examples. The meta-heuristic hybrid rice optimization (HRO), which has been effectively used in BS, roughly divides its population into three groups with an equal number of members based on self-equilibrium and symmetry. However, the main HRO has significant limits when it comes to the local search for better alternatives, and this could lead to the missing out of a good option. For BS, a modified HRO based on a differential evolution operator and an opposition-based learning technique is proposed in [43].

According to the authors' formulation in [44], the HSI BS problem evaluates the performance of every sparse band combination. It is a multitask sparsity pursuit problem. It is crucial to find the best answer to this issue as it can enhance how well image processing algorithms function. An attention module and an autoencoder are combined to form a neural network module that Dou et al. [45] called an attention-based autoencoder model. The informative band subset is chosen by the attention module, and the autoencoder only uses this subset to reconstruct the input data. When there is a lot of noise in the input data, this method can help an autoencoder perform better.

In [46], a multicriteria semisupervised model is developed for the selection of hyperspectral picture bands. The model is

broken down into two separate tasks. The first task evaluates the amount of information and redundancy contained in the chosen bands using unlabeled samples, whereas the second task evaluates the discrimination of the chosen bands using examples that have been labeled. In order to optimize this model, a multitask optimization approach is developed to aggregate the data from the bands and expedite the search for viable bands. In [47], from the labeled data, signature patterns are extracted with minimum and maximum reflectance values for each class, which are then quantized. The quantization process is carried out repeatedly until distinct patterns are found for each class. Finally, to guarantee that the selected bands have the least possible redundancy, bands with the highest correlation and lowest variance are discarded.

Cao et al. [48] propose a supervised BS technique based on the local spatial information of the hyperspectral picture and the wrapper method in light of the special characteristics of HSIs. The suggested technique consistently outperforms the traditional wrapper method by making use of the data from both labeled and unlabeled pixels in the HSI. In [49], unsupervised HSI BS using band grouping and adaptive multigraph constraints was suggested. When using a band grouping strategy to create a global similarity matrix, the problem of disregarding substantial correlations across neighboring bands is resolved. In contrast to prior research work that was limited to fixed graph restrictions, this approach creates a global similarity matrix by dynamically altering the weight of the local similarity matrix.

III. MATERIALS AND METHODS

The methodology section includes descriptions of the suggested approach, algorithms used, and the datasets. The method of merging algorithms served as the foundation for the BS strategy employed in this investigation. The evolutionary algorithm, one of the most widely used search techniques, has been integrated with artificial intelligence, particularly deep learning, to choose the right number of bands and reduce the dimensions for issues, such as the proposed method for classifying HSI. The convolution network as a fitness function, implemented in the evolutionary algorithm, is recommended as a mechanism for getting suitable and optimal bands in the processing of HSI. This mechanism is more effectiveness of genetic operators.

The structural alteration and transformation of the suggested neural network model in the proposed method have been reviewed and assessed in the Results and Discussion section with the goal of increasing accuracy and decreasing processing speed.

A. Convolution Neural Network Embedded GA (CNNeGA—Proposed Method)

Reducing the dimensions of input data is effective in many important problems and processes. Dimension reduction as a preprocessing issue in HSI processing is also done by finding appropriate subbands, which will be efficient in the next steps and subsequent processing of related issues. To find suitable subbands, we suggest using a GA and a convolutional neural network with 3-D layers (3D-CNN), which are embedded in the GA as a fitness function, which is responsible for classifying

TABLE I
PSEUDOCODE OF CNNeGA (PROPOSED METHOD)

1: Set parameters
2: Generate the initial population of Sub-bands
3: While $I < \text{Max Iteration and Best Fitness}$
4: Fitness calculation (Classification Accuracy of selected sub-bands using 3D-CNN)
5: Apply the Selection operator (ranking and selecting the sub-bands)
6: Apply the Crossover operator for the selected sub-bands with Check Parent
7: Apply the Mutation operator for the selected sub-bands with Check Parent
8: Generate the new population (population of selected sub-bands)
9: End While
10: Print Best Fitness and Optimal Output(Best Sub-band)

the input data. A population of subbands of the main HSI and the images produced from these subbands is applied to the proposed model, and the results of the classification are used for ranking and choosing suitable subbands. Finding the suitable subband and finally reducing the dimensions as the final result, using this combination, have been done with better and acceptable success. The combination of the GA with the designed 3D-CNN, which we named this combination CNNeGA, using the GA a population of solutions is generated. Then, each solution is applied to the convolutional neural network to evaluate its effectiveness and the result of the classification is examined to evaluate that solution. Better solutions are used with existing successful solutions by applying common operators in the GA (selection, crossover, mutation, etc.) to produce new generations and superior solutions. This process is repeated until the best solution is found and proposed. The pseudocode and its implementation steps are also shown in Table I, and the structure and flowchart of the proposed method (CNNeGA) are shown in Fig. 2.

One of the key advantages of 3D-CNNs is their ability to quickly identify patterns in 3-D data. This is important because many real-world problems involve 3-D data, such as recognizing objects in images or videos. Traditional neural networks are not suitable for this type of data and can often struggle [50], [51], [52], [53], [54]. However, 3D-CNNs are specifically designed to handle this type of information and can achieve much better results than traditional networks. 3-D convolutional neural networks are the future of machine learning and can learn from data much faster than traditional neural networks [55]. Traditional neural networks use the process of back-propagation to learn from data. This process can take a long time, especially when a lot of data is being processed. 3D-CNN uses a different process called convolutional layers, which can significantly speed up the learning process. Convolution layers divide the input data into small pieces and then process them in parallel. This allows the network to learn faster and more effectively than traditional networks. In addition, 3D-CNNs can more accurately represent complex patterns in datasets [56], [57]. The complexity layer, composite layer, and fully connected layer are some of the layers that make up the 3D-CNN, a CNN multilayer neural network.

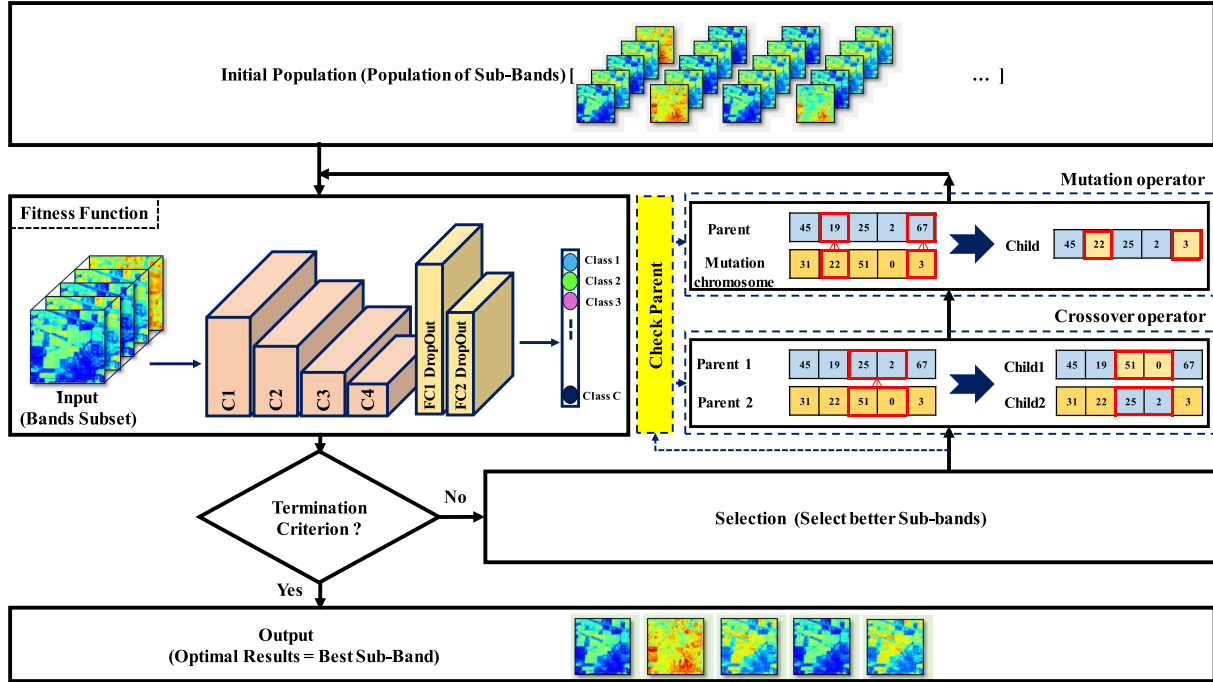


Fig. 2. CNNeGA—proposed method (3-D CNN embedded in GA) block diagram.

The first CNN model's convolution layer is used to execute the convolution operation on the input data. The 2D-CNN models can only extract spatial information, hence, the 3D-CNN-based model can not only extract spatial characteristics but also derives a spectrum for such models, which is why it is chosen and used. In comparable circumstances, models based on spatial extractors perform significantly better than spectral-spatial feature extractors.

In remote sensing applications, 3D-CNN can be used efficiently as these data include spectral and temporal features [58]. The ReLU activation function is used in the middle layers of the convolution network. ReLU has better behavior and is the most recommended activation function [59], [60]. At the ReLU output, a negative value is filtered to zero. The fast convergence of ReLU is what enables the activation function of CNN layers. This component allows two jumpers for reliable network operation. The problems of vanishing gradients are significantly reduced and saturation is avoided. The structure of the 3-D convolutional neural network designed and embedded in the GA algorithm, which is used here as a fitness and classifier function, and how to apply the data as input and output are shown in Fig. 3.

Each of the test data is placed in 3-D form (cube) in the input layer, and as mentioned before, while passing through each 3-D convolution layer, they gradually pass through the ReLU function. Also, the dropout technique is used to increase the efficiency and accuracy of the network before using the output layers. Regarding the input data to the network, it is necessary to explain that the spectral channels in each subset of the band applied to the convolution network are arranged in ascending order. As a result, data from the same channel are sorted into the same row. Therefore, the spectral background correlation will

remain stable in any selected band subset [61]. When 3D-CNN is applied to the hyperspectral input, the results will be as follows:

$$Z(x, y, b) = F \left(\sum_{i=1}^I \sum_{j=1}^J \sum_{d=1}^K (I(x+i, y+j, d + k) * N(i, j, k) + b) \right) \quad (1)$$

where K is demonstrated as the spectral size of the 3-D kernel.

B. Genetic Algorithm

GAs effectively solve many optimization problems, including constrained optimization problems, scheduling problems, routing problems, design optimization problems, and extracting suitable features. They are also more efficient than other search methods, such as hill climbing or simulated annealing, because they can be implemented in software or hardware for real-time applications, such as control systems or machine learning [62], [63]. Fig. 4(a) displays the GA's basic building blocks. The GA starts with a population of potential solutions or chromosomes. Then, it evaluates the fitness of each chromosome using fit criteria. Chromosomes that are better than others are more likely to reproduce and produce chromosome offspring. This process is repeated until a satisfactory solution is found. GAs have been used in various fields, such as image processing and HSI BS. In HSI, the goal is often to find an optimal set of bands that can be used for classification or feature extraction tasks. A GA-based approach is effective in quickly and efficiently finding good binding combinations. The main steps of using GA to select the HSI band are as follows.

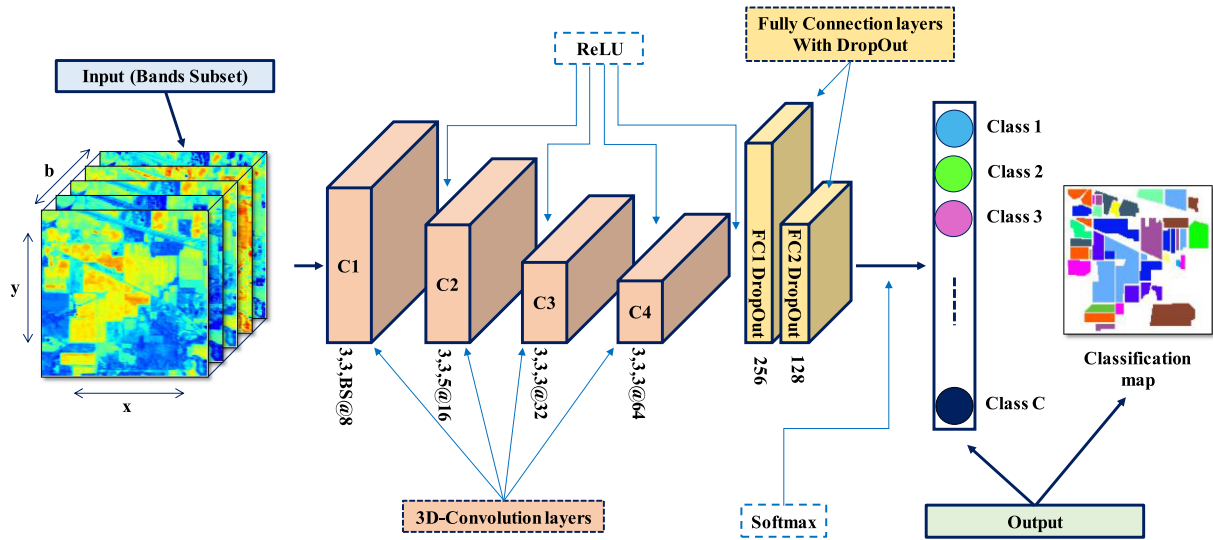


Fig. 3. 3D-CNN structure and how to apply data input and receive output data to the 3D-CNN structure.

- 1) Create the initial population.
- 2) Select the fitness function.
- 3) Evaluate the fitness of each chromosome.
- 4) Reproduction of successful chromosomes.
- 5) Mutation of failed chromosomes.
- 6) Repeat until reaching the convergence criteria [64], [65], [66].

The fundamental building blocks of a GA are chromosomal representation, fitness function, and genetic operators, including crossover, mutation, and selection. After randomly initializing a population of chromosomes, new chromosomes are produced by updating the genes on the pre-existing chromosomes in accordance with the fitness function. The best chromosomes in the population are selected for reproduction to produce new offspring, with some crossover between them to create diversity. This mutation occasionally introduces new solutions to the population. The GA is iterated until a satisfactory solution is found or the termination criteria are met [67]. The structure of this algorithm and the details of the mutation and crossover operators used in the basic GA are shown in Fig. 4 for a better understanding.

To generate a new population, the operators used in the GA used in this article are as follows, and their performance is shown in Fig. 4(b). The crossover operator is of the two-point type and the mutation operator is also selected from the flipping type. As can be seen in Fig. 5, in the crossover operator, a part of the selected subband is combined with another part of the selected subband and a new chromosome is produced. In the flipping mutation operator, two selective bands are shifted with each other and a new chromosome is produced. For the selection block in the GA, the rotating wheel selection operator is also used.

C. Check Parent

Genetic operators (crossover, mutation) are used to generate and improve new chromosomes. The repetition and commonality of some elements (bands) in parent chromosomes (subbands)

during population production is a well-known and unavoidable problem. This defect will make the process of selecting subbands difficult. Also, this causes the ineffectiveness of crossover and mutation operators in the GA. To overcome this problem, our proposed solution is to add a box to check parent chromosomes before entering the crossover or mutation operator. Fig. 4(a) and (b) shows this box and its location. Fig. 4(c) also depicts the proposed solution's details and implementation steps. After evaluating the initial population, P1 and P2 are selected as parental chromosomes. These parent chromosomes are selected in the crossover operator to generate new chromosomes C1 and C2, and similarly in the mutation operator, P, MC, to generate C. This process is repeated if P1, P2 for crossover and P, MC for mutation are not equal. If the constituent elements of the selected chromosomes ($P1 \neq P2$, $P \neq MC$) are not the same, therefore, none of the genetic operators will be applied to them. First, it will be explained how to implement the steps of the proposed solution, before the crossover operator, and similarly, it will be done about the mutation operator. In step 1, a list of common bands in P1 and P2 is prepared; the length of this list is considered to be L . If the length of L is greater than $1/2 b$ (b is the length of P1 or P2) (that is, if the number of bands in the list of L is more than half of the bands of P1 or P2), then the bands in the list of L with the probability of L/b are directly placed in C1 and C2, and the crossover operator is applied to P1 and P2. If this possibility ($P(L/b)$) fails and a band in P1 or P2 has not been repeated several times, the process is terminated by applying the crossover operator to P1 and P2. Because the initial population is produced at randomly, the presence of a repetitive band in some chromosomes is obvious. In the second step, a list of repeated bands in P1 along with their number, which is M , is prepared. Each repeated band, along with its additional number, is colored in Fig. 4(c) and Step 2. In this step, the extra bands are removed from P1. In the following, each of the additional repeated bands is replaced with the probability $P(\tau)$, which is equal to $1/K$ from the list of all bands except the repeated bands. This step is repeated similarly for P2. After this step, the crossover operator is applied to P1 and P2. This process and its stages are repeated

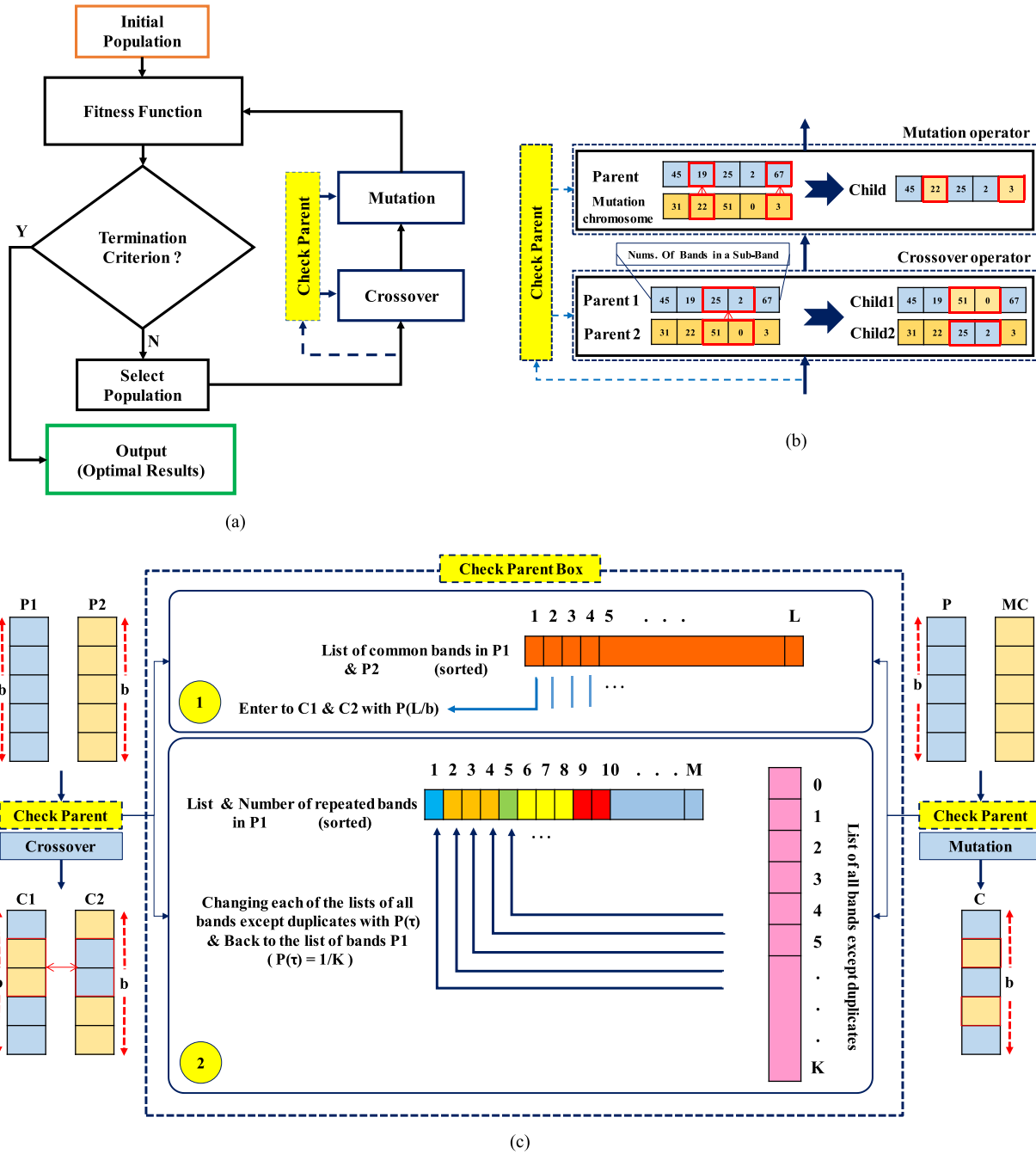


Fig. 4. (a) Basic structure of the GA with check parent box. (b) Crossover and mutation operators. (c) Check parent box.

in the same way before the selected chromosomes enter the mutation operator. This solution was presented to prevent the repetition of a band and maintain effective common bands in each band category in the generated population. It also avoids the ineffectiveness of crossover and mutation operators in the face of similar and repeated bands. Maintaining effective and suitable bands during this process on each selected and produced chromosome is also one of the advantages of this proposed solution. All operations in this process deal with the ID of each of the bands, not the content of the bands. As a result, it does not have an influential role in computational complexity. Table II shows the pseudocode of the parent box check.

D. Dataset

The efficiency of the suggested BS strategy for categorizing HSI land cover is assessed using three sets of publicly accessible HSI remote sensing data. These are the key characteristics of this dataset.

The first HSI dataset was gathered by AVIRIS sensor over the Indian Pines (IP) test site in North-Western Indiana and consists of 145×145 pixels and 224 spectral reflectance bands in the wavelength range $0.4-2.5 \times 10^{-6}$ m. The IP setting is made up primarily of agriculture, with the remaining third being either forest or other types of perennial forest vegetation. Two

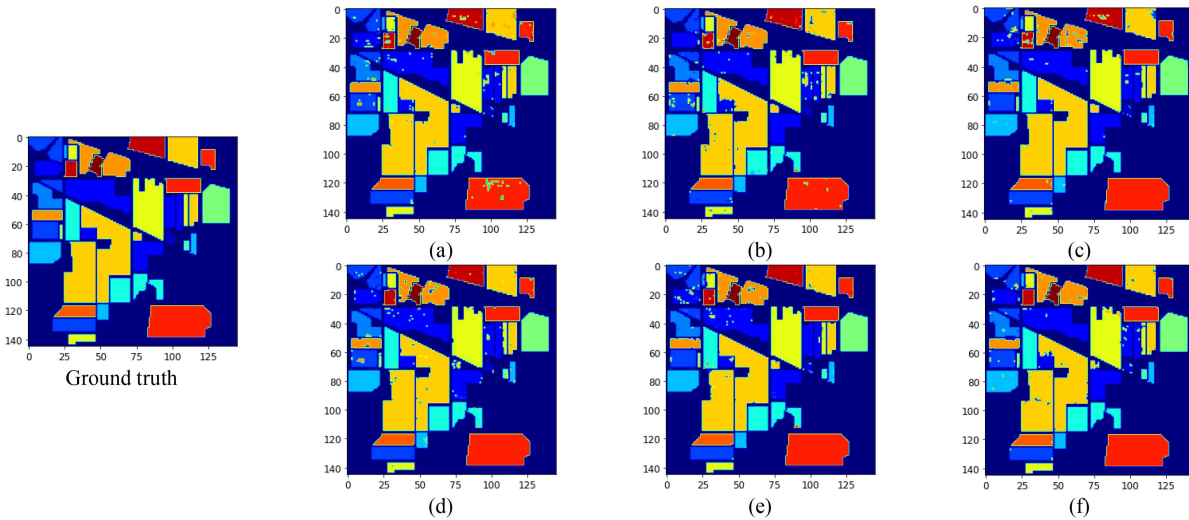


Fig. 5. Ground truth data and classification maps for dataset show the selection of (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected for IP datasets.

TABLE II
PSEUDOCODE OF CHECK PARENT BOX

<p>1: Check $P1 \neq P2, P \neq MC$ 2: If Positive, continue, otherwise Go to End</p> <p>Step 1: 1: Make a list of common bands (L-list) 2: if $L \geq \frac{1}{2} b$, L comeback to P1, P2 with P(L/b) 3: if P1 & P2 have not repeated bands Go to Step2-3.</p> <p>Step 2: 1: Make a list of repeated bands (M-list) 2: Replace repeated bands with $P(\tau) = 1/K$ for (K-list) 3: Go to Crossover Operator</p> <p>(Similarity for Mutation Operator) End</p>
--

TABLE III
INDIAN PINES DATA SET (FALSE-COLOR IMAGE, GROUND-TRUTH, SAMPLES)

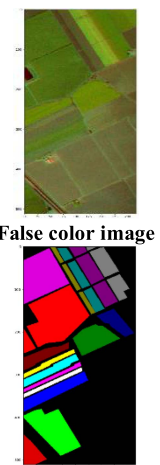
#	Classes Name	Samples
0	Background	56975
1	Broccoli_Green_Weeds_1	2009
2	Broccoli_Green_Weeds_2	3726
3	Fallow	1976
4	Fallow Rough Plow	1394
5	Fallow Smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes Untrained	11271
9	Soil Vineyard	6203
10	Com Senesced Green Weeds	3278
11	Lettuce_Romaine_4wk	1068
12	Lettuce_Romaine_5wk	1927
13	Lettuce_Romaine_6wk	916
14	Lettuce_Romaine_7wk	1070
15	Vineyard Untrained	7268
16	Vineyard Vertical Trellis	1807

significant dual-lane highways, a rail line, some low-density dwellings, other developed features, and minor roads are all present. Since the scene is June, some of the crops, such as maize and soybeans, are just starting to grow and have less than 5% coverage. Sixteen categories of ground truth can be used, although not all of them apply to every situation. By eliminating the bands [104-108], [150-163], and 220 that cover the water absorption region, we have also lowered the number of bands to 200. You can access IP data on Pursue’s University MultiSpec website. Table III displays the data and representation for this dataset. The 224-band AVIRIS sensor was used to collect the second HSI dataset, which is displayed in Table IV, as a representation of a false color image as well as a real ground image and its sample class information. This dataset has a high spatial resolution (pixels 3.7 m) and was taken in the Salinas Valley (SA) of California. The area covered includes 512×217 . We eliminated 20 water absorption bands, in this case, bands [108-112], [154-167], and 224, much as we did with

the IP landscape. The only form of this image that was stored in the sensor was radiance data. Vegetables, barren soils, and grape lands are included in this. Salinas’s dataset consists of 16 classes. The University of Pavia dataset (PU) from Table V is the third HSI dataset. It was collected by reflective optical imaging spectroscopy in the urban area surrounding the University of Pavia in northern Italy. Nine separate classes, each representing various urban terrestrial items, are taken into account in the actual earth map. Each band has a size of 610×340 pixels with a spatial resolution of 1.3 m per pixel. It was captured between the wavelengths of 0.43 and 0.86×10^{-6} m. There are 115 spectral bands in this HSI. Twelve extremely noisy bands were eliminated before the experiment, leaving 103 bands that may be used for research. The fourth HSI dataset shown in Table VI was obtained. The NSF-funded Center at Airborne Laser Mapping acquired the data from the University of Houston campus and

TABLE IV
SALINAS DATA SET (FALSE-COLOR IMAGE, GROUND-TRUTH, SAMPLES)

#	Classes Name	Samples
0	Background	56975
1	Broccoli_Green_Weeds_1	2009
2	Broccoli_Green_Weeds_2	3726
3	Fallow	1976
4	Fallow Rough Plow	1394
5	Fallow Smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes Untrained	11271
9	Soil Vineyard	6203
10	Com Senesced Green Weeds	3278
11	Lettuce_Romaine_4wk	1068
12	Lettuce_Romaine_5wk	1927
13	Lettuce_Romaine_6wk	916
14	Lettuce_Romaine_7wk	1070
15	Vineyard Untrained	7268
16	Vineyard Vertical Trellis	1807




False color image

Ground truth

TABLE V
UNIVERSITY OF PAVIA DATA SET (FALSE-COLOR IMAGE, GROUND-TRUTH, SAMPLES)

#	Classes Name	Samples
0	Background	56975
1	Broccoli_Green_Weeds_1	2009
2	Broccoli_Green_Weeds_2	3726
3	Fallow	1976
4	Fallow Rough Plow	1394
5	Fallow Smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes Untrained	11271
9	Soil Vineyard	6203



False color image

Ground truth

the neighboring urban area. The Houston (H) dataset consists of 144 spectral bands in the region of 380 to 1050 nm and its spatial resolution is 2.5 m. The average height of the sensor above the ground was 5500 ft. Houston's dataset consists of 16 classes and 349×1905 pixels. A summary and initial information about the four datasets used are provided in Table VII.

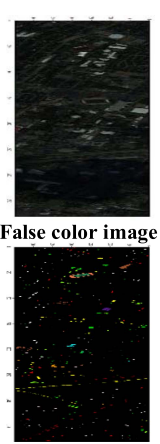
IV. RESULTS AND DISCUSSION

In this section, several comparative experiments have been planned, developed, and used on four HSI datasets in this part. The outcomes of these comparisons have been assessed and analyzed using six competing methodologies. Analyzes and discussions are done in several categories, which are as follows.

- 1) Parameters and settings are required in the proposed algorithm and method.
- 2) Comparison of the accuracy obtained from the results of the proposed method with six competing methods.

TABLE VI
UNIVERSITY OF HOUSTON DATA SET (FALSE-COLOR IMAGE, GROUND-TRUTH, SAMPLES)

#	Classes Name	Samples
0	Background	649986
1	Grass Healthy	1251
2	Grass Stressed	1254
3	Grass Synthetic	697
4	Tree	1074
5	Soil	1242
6	Water	325
7	Residential	1268
8	Commercial	1244
9	Road	1252
10	Highway	1227
11	Railway	1235
12	Parking Lot1	1233
13	Parking Lot2	469
14	Tennis Court	428
15	Running Track	660



False color image

Ground truth

TABLE VII
DETAILS ON DATASETS

Dataset	Resolution	Pixels background object	Bands	Available Band	Used Class
IP	145*145	21025	220	200	16
		10776 10249 111104			
SA	512*217	56975	224	204	16
		54129 2207400			
PU	610*340	2164624	115	103	9
		42779 664845			
H	349*1905	649986	144	144	16
		14859			

- 3) Distribution of selected and number of bands in each tested dataset.
- 4) Selected ideal and recommended bands.
- 5) The results of the classification of images with the real earth image using recommended bands.
- 6) The distribution of selected bands along with the spectrum of classes in each batch of experimental data.
- 7) Complexity analysis.
- 8) Evaluation and effectiveness of increasing layers of attention in performance.
- 9) Converting the 3D-CNN model to a spike neural networks (SNNs) and checking the results.

To show and compare the results of the proposed method fairly, three competing methods have been selected. These methods are selected from supervised categories, and named according to their references as follows.

- 1) BS_UQ (BS using quantization) [47].
- 2) LSI_BS (local spatial information BS) [48].
- 3) SLN_BS (single-layer neural networks BS) [41].

TABLE VIII
GA PARAMETERS

Max generation	50
Size of population	Num. of BS
Crossover probability	0.9
Initial mutation probability	0.02
Ultimate mutation probability	0.1

TABLE IX
3D-CNN PARAMETERS IN DIFFERENT BS

5 BS	913392	20 BS	12956712
10 BS	4927832	25 BS	16971152
15 BS	8942272	30 BS	20985592

The focus of this article is on the relative comparison of the results with the mentioned methods.

A. Implementation Details

In the experiment, the simulation and modeling process in this research has been implemented through Python. Using the Tensor-flow and Keras libraries, on the Collaboratory platform, as well as using GPUs with free access and suggested by this platform. The early stop strategy has been employed in the validation to avoid the model from becoming overfitting and to shorten the proposed method's execution duration. We set the Epoch number to the lowest value of 35 because the 3D-CNN 35 to 50 epoch range produces the maximum level of classification accuracy.

The ReLU and SoftMax activation functions were employed, respectively, in CNN's middle and output layers. The number of the selected band (BS) was considered as one of the set parameters of the first layer filter in the 3D-CNN as (3*3*BS) (see Fig. 3), which means that in all BS modes, the value with the desired number is considered for the selected band category. Also, that each iteration, 20% of samples are used for training the network, and the GA's iteration is set at 100.

B. Parameters and Settings

This section includes a list of the various settings and the number of parameters for the 3D-CNN in the proposed method. The different parameters and BS modes are shown in Tables VIII and IX.

These settings include the initial value of the parameters used in GA and the numerical number of parameters derived from the convolution network, which is included in the GA. Each of the operators used in the GA, including the ranking and selection of the optimal chromosome, crossover, and mutation, has a value as a probability, and the optimal value of this probability is recorded during the initial settings and used in the process of running the algorithm has taken. Also, the number of parameters or weights used in the convolutional neural network, in selecting different bands from 5 to 30 bands, is shown in these tables.

C. Performance and Accuracy Analysis

The level of accuracy attained in classifying the image derived from the search for the ideal bands chosen by the proposed approach is shown in the first evaluation of the methods taken into consideration this article. The accuracy curve is created by changing the chosen band number, b , which ranges from 5 to 30 with a five-point interval, and the average accuracy bar is created by averaging the categorization accuracy rates used as a whole. It is necessary to explain that in all the evaluations performed on the desired and tested data, the accuracy was obtained and the result of the evaluation of the classification accuracy for the full band mode is the same according to the number of calculations and the length of time. The accuracy obtained in the 30-band mode is considered and excluded from further evaluation in the selected band number set.

Table X contains information about the accuracy of image classification HSI and the numerical comparison of the obtained results with the reference methods. This table's best and highest accuracy values are related to the results obtained from the proposed CNNeGA method. In this table, the maximum and minimum values obtained by competitive methods can be seen. The bold black font corresponds to the maximum precision values in each of the selected bands. The differences in the value of accuracy of the competing method with the proposed method in all cases and tested data are about 6% to 21%. The evaluation of the obtained accuracies about testing the data related to the IP dataset shows that the maximum accuracy values obtained in the competing methods are related to the BS_UQ method and are between 0.85 and 0.9508. These values are lower than the proposed method in each selected band category from 1% to 5%. Also, the SLN_BS method has the lowest values obtained in this group of the IP dataset. The accuracy obtained by the proposed method (CNNeGA) is from 0.9085 to 0.9661. The accuracies obtained from testing the data related to the second SA dataset also show that the maximum accuracy values obtained in the competing methods are related to the BS_UQ method and are between 0.922 and 0.972. The accuracy values are lower than the proposed method in each selected band category from 1% to 3%. In this group, the LSI_BS method also has the lowest values obtained. The accuracy obtained by the proposed method (CNNeGA) is from 0.93007 to 0.982. The accuracies obtained in the third group, the test of the PU dataset also show with a slight difference that the maximum accuracy values obtained in the competing methods are related to the BS_UQ and SLN_BS methods and are between 0.906 and 0.97, of course, the SLN_BS method is in the category bands 15, and 20 has the highest amount of accuracy obtained. The accuracy values are lower than the proposed method in each selected band category from 2% to 8%. Also, the LSI_BS method has the lowest obtained values. The accuracy obtained by the proposed method (CNNeGA) is from 0.9807 to 0.9973.

The accuracies obtained from testing the data related to the last dataset H also show that the maximum accuracy values obtained in competing methods are related to the BS_UQ and SLN_BS methods and are between 0.892 and 0.9583, although the SLN_BS method is in the category bands 5, and 15 has the

TABLE X
ACCURACY OF METHODS IN THE CLASSIFICATION OF SUBBANDS HSI

Dataset	Methods	5 BS	10 BS	15 BS	20 BS	25 BS	30 BS
(IP)	SLN_BS	0.7061	0.7203	0.7331	0.7468	0.7481	0.75
	BS_UQ	0.85	0.908	0.937	0.9403	0.9491	0.9508
	LSI_BS	0.768	0.78	0.792	0.803	0.809	0.813
Salinas (SA)	CNNeGA	0.9085	0.94003	0.95014	0.9505	0.9658	0.9661
	SLN_BS	0.905	0.9086	0.9103	0.9119	0.9106	0.9109
	BS_UQ	0.908	0.9164	0.937	0.9481	0.9503	0.972
	LSI_BS	0.879	0.884	0.89	0.8955	0.896	0.8967
Pavia University (PU)	CNNeGA	0.93007	0.9573	0.9668	0.9728	0.9809	0.982
	SLN_BS	0.8895	0.9016	0.9207	0.9461	0.9482	0.9519
	BS_UQ	0.906	0.92	0.9125	0.9325	0.952	0.97
	LSI_BS	0.773	0.8	0.806	0.809	0.813	0.818
Houston (H)	CNNeGA	0.9807	0.9834	0.9927	0.9948	0.9949	0.9973
	SLN_BS	0.892	0.9104	0.9341	0.9372	0.9418	0.9437
	BS_UQ	0.87	0.9120	0.939	0.945	0.952	0.9583
	LSI_BS	0.714	0.746	0.735	0.751	0.75	0.762
	CNNeGA	0.9206	0.9438	0.9573	0.9717	0.9749	0.9807

highest amount of accuracy obtained. The accuracy values are lower than the proposed method in each selected band category from 2% to 12%. In this group, the LSI_BS method also has the lowest values. The accuracy obtained by the proposed method (CNNeGA) is from 0.9206 to 0.9807. The proposed method as well as the SLN_BS method are class-based and therefore based on classifier training. Of course, this issue in the proposed method is due to the use of a deep network based on CNN layers, which has higher accuracy in teaching and predicting the class. The SLN_BS method has an approach based on the use of machine learning. It seems that the approach of using a deep network on CNN in this problem is good and has the highest level of accuracy in the obtained results. But the SLN_BS method has more speed and less computational complexity than the proposed method. In the LSI_BS method, accuracies have been obtained using the SVM classifier. SVM classifier is one of the most common algorithms used in the classification of many problems and is the basis for comparing the proposed methods with this method. Of course, here too, the accuracy of the proposed method was proven higher than the LSI_BS method due to the use of a deep neural network as a classifier. This method has average computational complexity and speed between the proposed method and the SLN_BS method.

The level of accuracy attained in classifying the image derived from the search for the ideal bands chosen by the proposed approach is shown in the first evaluation of the methods taken into consideration in this article. The accuracy curve is created by changing the chosen band number, b , which ranges from 5 to 30 with a 5-point interval, and the average accuracy bar is created by averaging the categorization accuracy rates used as a whole. It is necessary to explain that in all the evaluations performed on the desired and tested data, the accuracy was obtained and the result of the evaluation of the classification accuracy for the full band mode is the same according to the

number of calculations and the length of time. The accuracy obtained in the 30-band mode is considered and excluded from further evaluation in the selected band number set. Table X contains information about the accuracy of image classification HSI and the numerical comparison of the obtained results with the reference methods. This table's best and highest accuracy values are related to the results obtained from the proposed CNNeGA method. The BS_UQ method has the closest results to the proposed method compared to the other two methods.

This method is based on feature extraction and the use of statistical methods to select a data-oriented band without user intervention. Although data labeling is used in this method, it does not depend much on this technique. But the BS_UQ method used 60% of the samples for training and 40% for testing the classifier to check the accuracy. Compared to the proposed method, more data (about three times) has been used for training and testing the classifier. In this method, an SVM classifier is also used. This method is almost equal to the LSI_BS method in terms of computational complexity. In general, the proposed method has better performance due to the use of less number of training samples, higher stability, and accuracy of classification in the face of various data. The proposed method shows promising results compared to the other mentioned methods in that the classification accuracy of PU and IP datasets reaches more than 99% and 98% in some cases. However, less than 97% was obtained for some datasets, this may be because the data (classes) are larger and imbalanced, have a relatively lower spatial resolution, and also have a much larger number of cover classes. The selection of bands efficiently establishes the resolution between various land cover classes without data loss or spectral distortion. Figs. 5–8 show the four datasets IP, SA, PU, and H separately so that you can visually assess the effectiveness and classification accuracy of the selected band set of 5 to 30 bands using the advised CNNeGA method as well

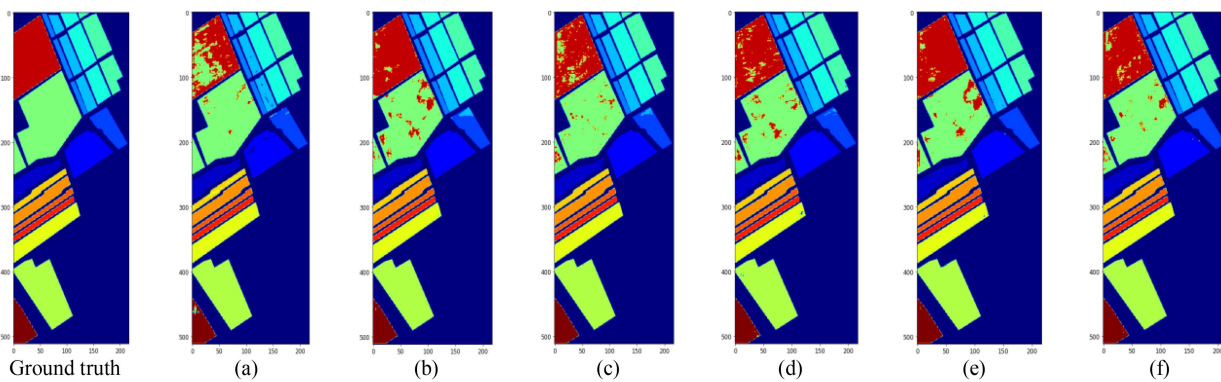


Fig. 6. Ground truth data and classification maps for dataset show the selection of (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected for SA datasets.

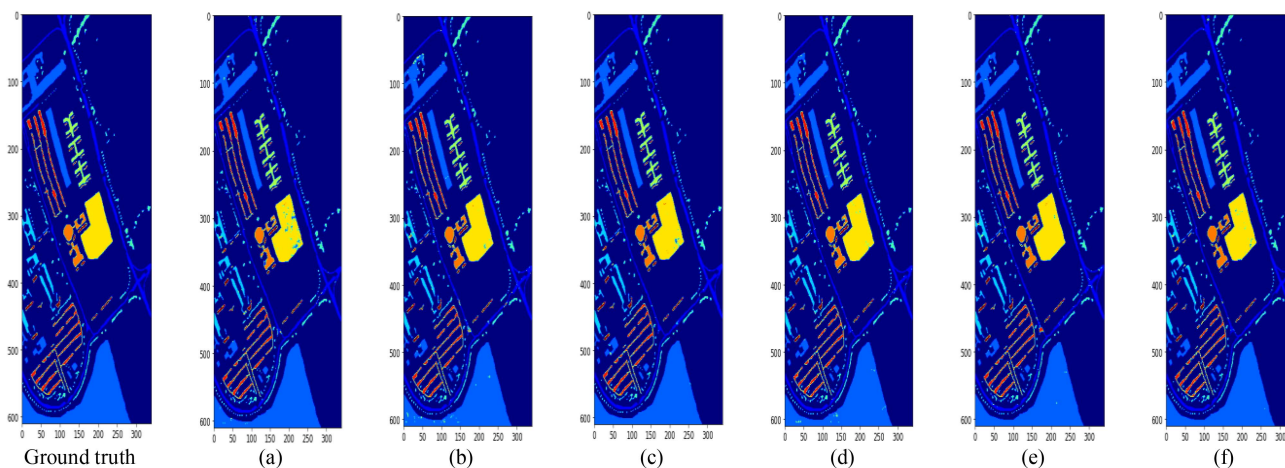


Fig. 7. Ground truth data and classification maps for dataset show the selection of (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected for PU datasets.

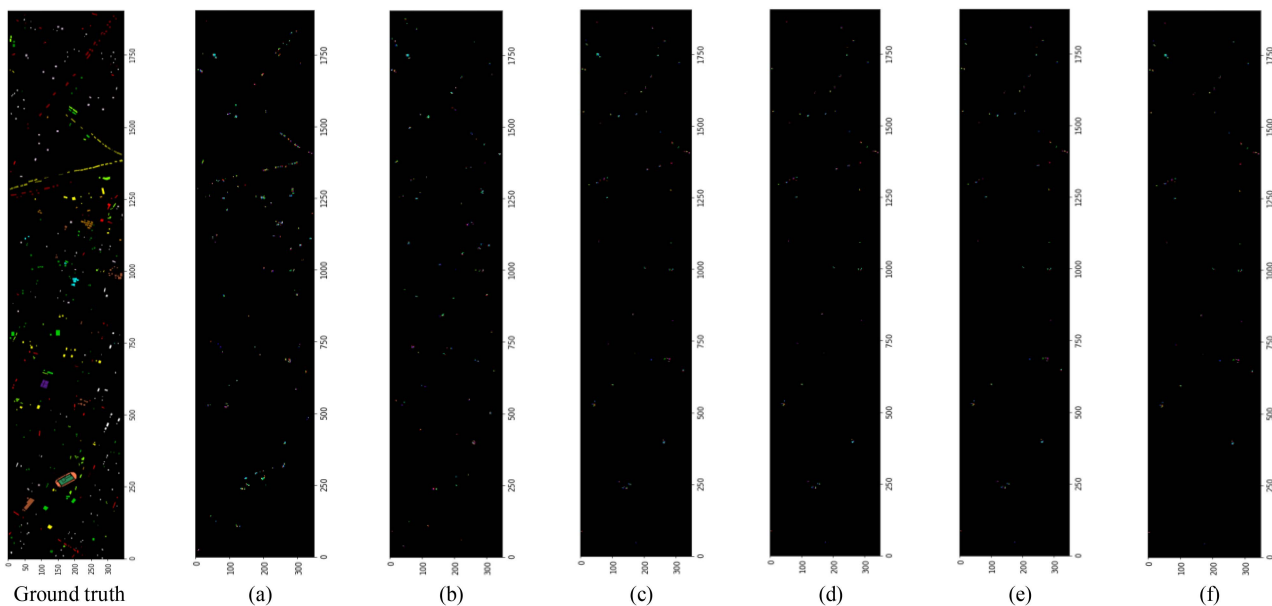


Fig. 8. Ground truth data and classification maps for dataset show the selection of (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected for H datasets.

TABLE XI
IDEAL AND RECOMMENDED BANDS IN DIFFERENT SUBBANDS PER THE HSI DATASET

Dataset	The list of bands in different sub-bands	Classification Accuracy	
IP	5 BS	[23,36,81,118,180]	0.9085
	10 BS	[8,31,45,55,68,85,115,141,155,165]	0.94003
	15 BS	[2,13,15,21,32,61,70,107,134,142,164,165, 181,182,196]	0.95014
	20 BS	[9,14,15,27,41,45,73,75,79,85,96,104,115,125,144,147,155, 165,181,191]	0.95057
	25 BS	[3,6,10,11,23,29,34,35,42,47,56,66,81,85,104,110,141,142,145,155,161,172,181,189,199]	0.9658
	30 BS	[2,9,12,23,34,36,41,45,56,68,85,96,98,100,104,114,118,121,128,129,134,145,155,163,166,177,181,186,187,191]	0.9661
All Bands	All Bands = 200 bands	0.9661	
SA	5 BS	[7,11,29,98,152]	0.93007
	10 BS	[13,22,43,82,108,116,150,164,197,203]	0.9573
	15 BS	[20,21,25,37,42,58,75,78,99,102,108,129,136,175,190]	0.9668
	20 BS	[0,5,11,23,36,54,68,85,91,95,104,106,116,117,120,150,169,171,172, 178]	0.9728
	25 BS	[6,26,39,51,59,65,68,75,94,99,106,109,116,120,130,133,136,138,141,147,150,155,161,175,200]	0.9809
	30 BS	[2,7,11,12,16,19,27,33,37,39,60,61,62,96,106,122,123,129,135,148,154,165,167,178,182,187,190,192,199,202]	0.982
All Bands	All Bands = 204 bands	0.982	
PU	5 BS	[20,36,64,71,91]	0.9807
	10 BS	[1,18,33,37,41,69,70,73,77,99]	0.9834
	15 BS	[0,3,10,11,15,20,35,40,48,61,66,74,85,88,97]	0.9927
	20 BS	[2,5,11,13,15,23,29,37,42,46,54,55,58,60,61,67,80,87,98,102]	0.9948
	25 BS	[0,4,11,12,17,22,36,38,42,46,48,51,52,55,59,60,61,69,76,78,79,84,89,96,101]	0.9949
	30 BS	[1,9,10,11,13,15,18,20,29,31,33,38,40,42,45,55,56,58,59,62,68,72,75,77,82,86,92,94,98,101]	0.9973
All Bands	All Bands = 103 bands	0.9973	
H	5 BS	[7,51,60,118,130]	0.9206
	10 BS	[7,38,42,75,88,101,102,111,115,121]	0.9438
	15 BS	[0,13,19,41,43,60,70,84,95,106,114,121,132,140,143]	0.9573
	20 BS	[0,6,11,12,15,31,35,45,54,56,70,84,89,93,104,107,110,128,141,143]	0.9717
	25 BS	[0,6,11,13,31,35,38,45,46,48,51,55,59,60,61,69,70,84,89,93,96,118,131,141,143]	0.9749
	30 BS	[0,7,10,11,12,13,20,29,31,33,38,45,54,55,56,58,59,62,68,70,84,88,94,96,101,106,110,121,130,140,143]	0.9807
All Bands	All Bands = 144 bands	0.9807	

as the variation between the prediction of the obtained images and the actual image.

Separately, the classification maps in this method and the ground truth in the IP dataset, including 16 feature categories in Fig. 5(a)–(f), in the Salinas dataset with 16 feature categories in Fig. 6(a)–(f), the Pavia University dataset, including 9 features, are shown in Fig. 7(a)–(f), and the Houston dataset, including 9 features, are shown in Fig. 8(a)–(f). As shown in Fig. 5, the proposed method gradually achieves better classification results in four different datasets by increasing the number of selected bands.

D. Band Selection

As shown in Table X, the proposed method gradually achieves better classification results in different datasets by increasing the number of selected bands. Figs. 5–8 show the accuracy, quality, and performance of the proposed method as it progresses toward increasing the number of bands in each desired image in each of the selected bands and improving the classification.

Table XI shows the proposed and ideal band using the CN-NeGA method tested and displayed on each of the datasets separately. In addition to displaying the selected bands, the

accuracy values obtained in the classification of each category of bands have also been re-entered. The display and distribution of the ideal bands selected in the CNNeGA method in the corresponding Fig. 9, and for each of the IP, SA, PU, and H test data separately and after the table of recommended bands for each, have been shown. If the selected bands contain too much redundant information, test data with this method are not suitable for classification tasks. First, we investigated the distribution of the selected bands to analyze the additional information in the selected bands with the proposed BS method. The selected and recommended groups in each category, from 5 to 30, are shown in different colors in Fig. 9. The uniform distribution of the selected bands, which are chosen from the entire spectrum of bands, serves as additional evidence that the suggested strategy was chosen well. Using this method to choose a band has several advantages, including the straightforward implementation and coding of the method and access to resources, such as the Colab platform, which is available without charge. The selected bands and the class spectrum curves from four different datasets are shown in Figs. 10–13, respectively, in each group of bands from 5 to 30 bands. In the shown graphic, each vertical line denotes where a certain band is located. The suggested BS approach may choose bands with low redundant information and

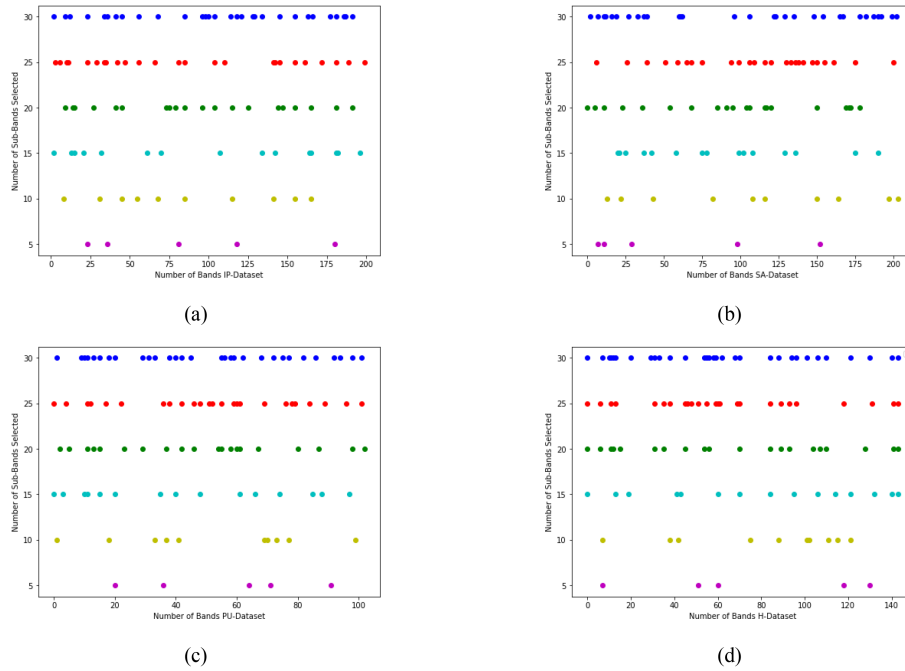


Fig. 9. Distribution of the optimal bands selected in the proposed method in (a) IP, (b) SA, (c) PU, and (d) H dataset.

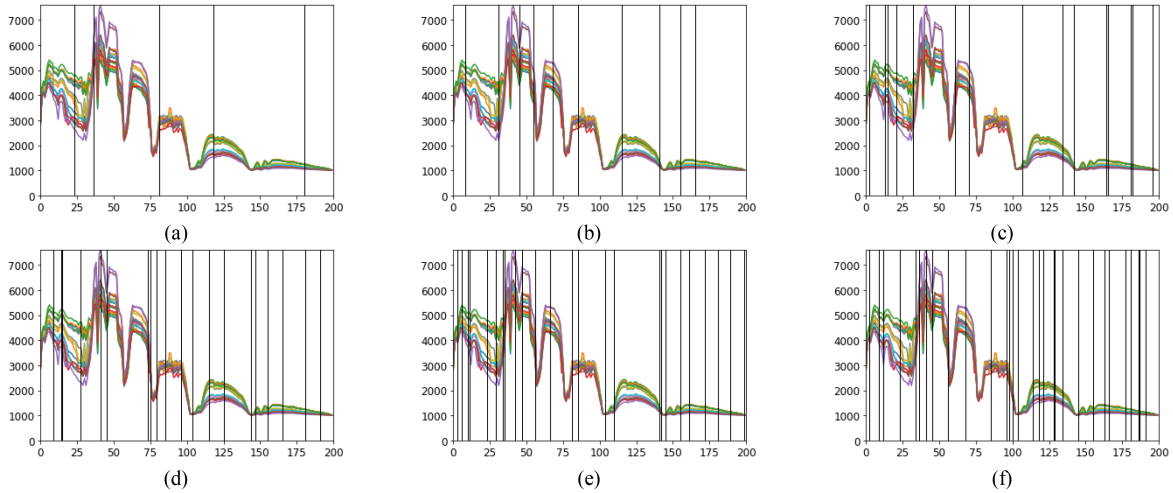


Fig. 10. Spectrum curves of the classes are included in the IP dataset. The vertical lines denote the different bands selected by the BS proposed method: (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected.

a distribution that is close to uniform, according to experimental data. Using the proposed method, a set of influential bands have been identified and spectral bands that do not contain useful information have been avoided. Fig. 9, with Figs. 10–13, is a complete visual representation of how to select a band using the proposed method. The evaluation criteria of the studied and proposed methods are the accuracy level obtained in the classification in each of the selected band categories and on each of the experimental data. From the results shown in Table X, and the accuracy obtained in classifying images into different categories in Figs. 5–8, and on the experimental data, we can see that our method achieves the highest performance in each

dataset and our hypothesis confirms that. The results, which are consistent even under investigation, show that there is a significant gap in performance between the proposed method and competing methods in the accuracy proposed method and competing methods in the accuracy achieved in band classification and selection. There are limitations in the implementation and the proposed method, which can mainly be pointed to the amount and time of calculations of this method, which is considered of the major limitations. The increase in the number of parameters of the CNN used, which increases in each of the selected bands, leads to an increase in the time and computation.

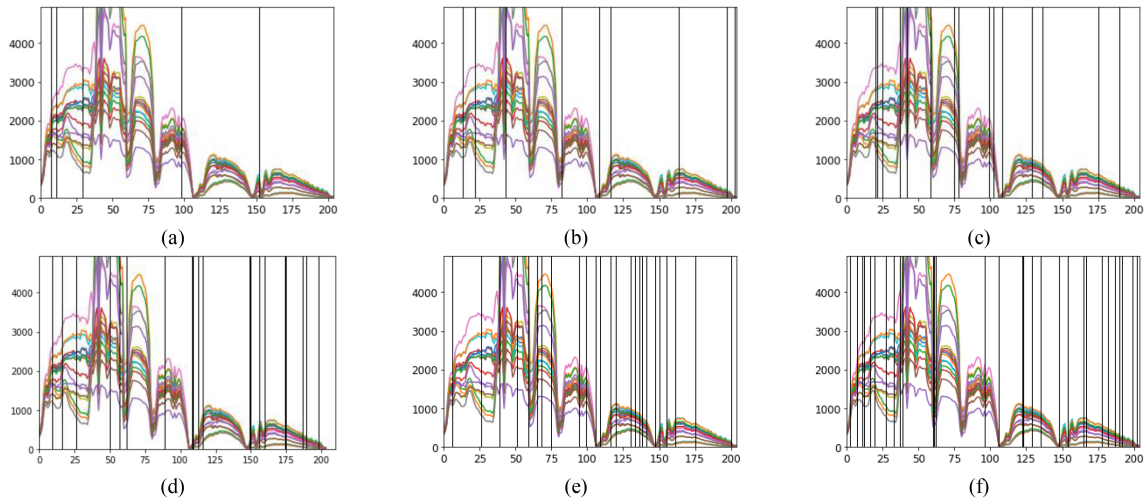


Fig. 11. Spectrum curves of the classes are included in the SA dataset. The vertical lines denote the different bands selected by the BS proposed method: (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected.

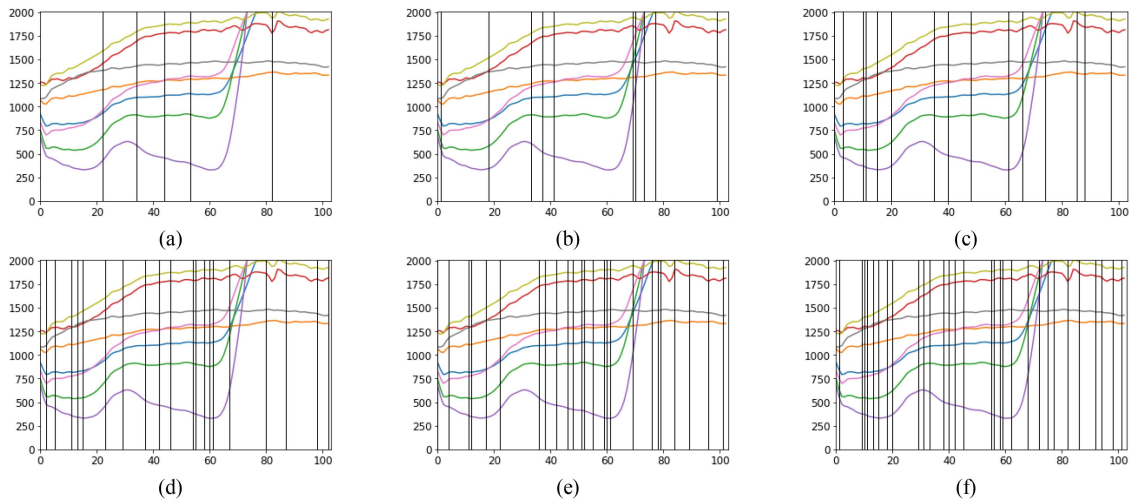


Fig. 12. Spectrum curves of the classes are included in the PU dataset. The vertical lines denote the different bands selected by the BS proposed method: (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected.

E. Effectiveness of Increasing Attention Layers in Performance

In this part, we pay attention to one of the ideas added to this article, that is, the level of effectiveness and adding attention layers to the 3D-CNN structure. The attention mechanism is used in many methods and increases accuracy, especially in classification methods. In the final part, after selecting and finding suitable bands (recommended band sets in Table XI), we evaluated the performance and classification accuracy of each band category by applying these bands to the new 3D-CNN network structure. This evaluation was done only to confirm the performance of the proposed method and retest the recommended subbands in each dataset. The performance evaluation showed that the accuracy of the classification is $\sim 3\%$ more and increased than the previous value. Although the evaluations show good performance, satisfactory results, and the increase in parameters (2 to 3 times the structure of the 3D-CNN network and the

parameters shown in Table IX), as a result, the computational complexity is one of the serious challenges. Fig. 14(a) shows the change in the 3D-CNN structure with the addition of the attention layer. In this structure, the attention layer is used in two forms: the spectral attention module and the spatial attention module, which are shown in Fig. 14(b) and (c). The weight of attention paid to each of the retrieved spectral and spatial features is altered by the spectral and spatial attention module, which is positioned after each 3-D CNN layer. Each channel generates additional channels with distinct information after being processed by various convolution kernels. Assume that each channel will have weights added for display. A stronger link and relationship between the channel and the important information imply a heavier weight. As a result, the appropriate channel needs to receive greater focus. Each feature channel's relevance is modeled by the spectral attention module, which subsequently boosts or suppresses them depending on the task.

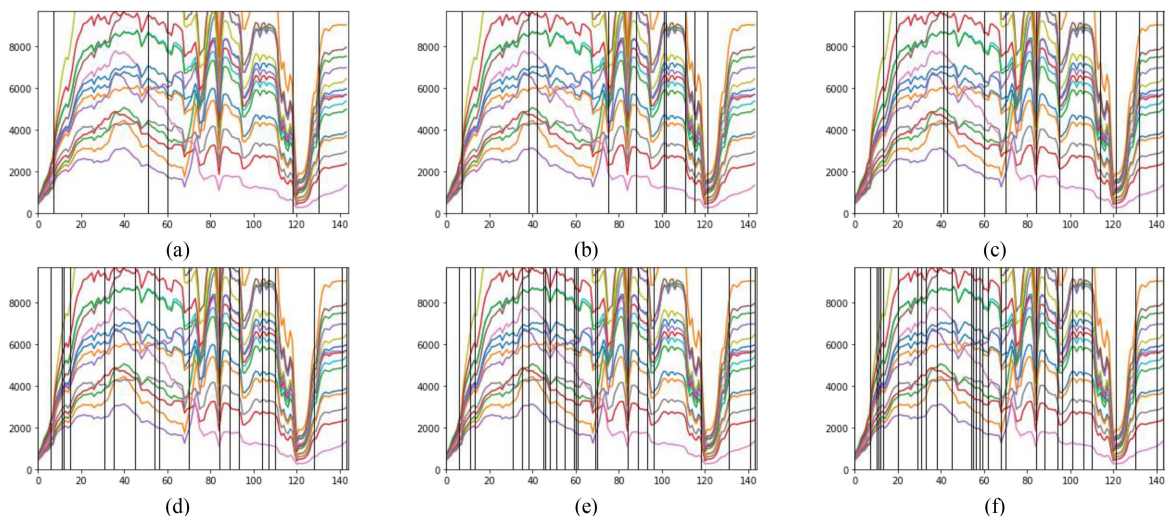


Fig. 13. Spectrum curves of the classes are included in the H dataset. The vertical lines denote the different bands selected by the BS proposed method: (a) 5 bands, (b) 10 bands, (c) 15 bands, (d) 20 bands, (e) 25 bands, and (f) 30 bands selected.

TABLE XII
ACCURACY OBTAINED IN THE CLASSIFICATION SUBBANDS RECOMMENDED HSI DATASET BY 3D-CNN

Dataset	Classification Accuracy of Recommended sub-bands						
	3D-CNN model	5 BS	10 BS	15 BS	20 BS	25 BS	30 BS
(IP)	With Att. Layers	0.9203	0.9518	0.9731	0.978	0.9827	0.9812
	Without Att. Layers	0.9085	0.94003	0.95014	0.9505	0.9658	0.9661
Salinas (SA)	With Att. Layers	0.938	0.9651	0.97	0.988	0.99	0.994
	Without Att. Layers	0.93007	0.9573	0.9668	0.9728	0.9809	0.982
Pavia University (PU)	With Att. Layers	0.998	0.998	0.998	0.999	0.999	0.999
	Without Att. Layers	0.9807	0.9834	0.9927	0.9948	0.9949	0.9973
Houston (H)	With Att. Layers	0.9528	0.9614	0.9704	0.9882	0.99	0.9913
	Without Att. Layers	0.9206	0.9438	0.9573	0.9717	0.9749	0.9807

A spatial attention map develops details by utilizing the link between the spatial features of the information in concentrated chunks. Spatial attention is distinct from spectral attention. In the structure of 3D-CNN with attention layers, there are second paths to extract the cube features of the input image. In this structure, one path is used to extract spectral features and the second path is used to extract spatial features. Finally, after connecting the output of these two feature extraction paths, the feature map obtained in the last layer of 3-D CNN (C4) from the global average pooling (GAP) module has been used to create the feature vector. In the networks that use the fully connected layer for classification, the output feature maps are given to SoftMax after joining each other, but in this method, a feature map is generated for each class after the last CNN layer. The GAP layer is added to the network. By adding a GAP layer on top of the feature maps, they benefit from the feature maps and the feature vector results are directly given to SoftMax. One of the characteristics of the GAP layer is that it is not more than a problem due to the lack of parameters for

optimization, and on the other hand, this layer is more resistant to local changes and more compatible with convolution networks [79], [80], [81], [82], [83], [84], [85], [86], [87]. Table XII shows the accuracy result obtained using the 3D-CNN network based on the attention layers, which shows the four datasets of IP, SA, PU, and H. It seems that in datasets where the similarity between classes (spectral similarity), such as on the SA dataset, the accuracy obtained by using the attention mechanism is not much different from the previous values.

F. Spike Neural Networks

SNNs are biologically plausible counterparts of artificial neural networks (ANNs). ANNs are usually trained with stochastic gradient descent, and spiking neural networks are trained with spike-timing-dependent plasticity. Training deep convolutional neural networks is a memory and power-intensive task. SNNs could potentially help in reducing power usage. There is a large pool of tools for one to choose to train ANNs of any size.

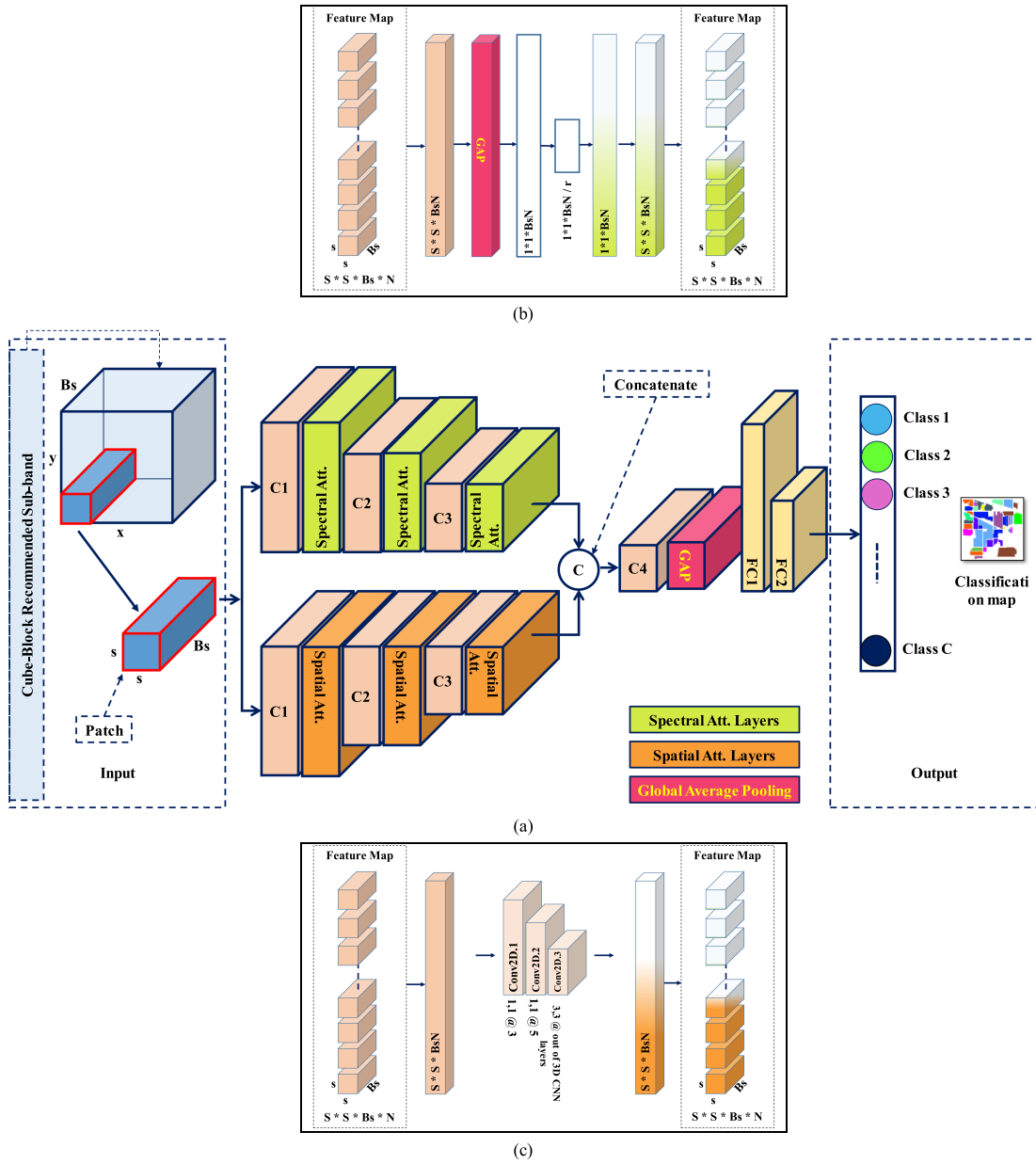


Fig. 14. (a) Architecture of the 3D-CNN with attention layers. (b) Spectral attention module. (c) Spatial attention module.

All the available tools to simulate spiking neural networks are geared toward computational neuroscience applications. SNNs promise that they are less computationally intensive and much more energy efficient because it runs asynchronously using spikes. SNNs have gained massive attention as a potential energy-efficient alternative to conventional ANNs due to their inherent high-sparsity activation. However, most previous SNNs methods use ANN-like architectures, which can provide optimal performance for the processing of binary information in SNNs. This section focus on implementing a deep spiking CNN. The 3D-CNN (proposed model) converts into an SNN network to implement our idea for providing a method. Of course, our approach in this section is only a reference to the conversion

method and some components and techniques in the use of SNNs [88], [89], [90], [91].

ANNs and SNNs can model the same types of network topologies, but SNNs trade the artificial neuron model with a spiking neuron model instead. The artificial neuron model much like spiking neurons operates on a weighted sum of inputs [see Fig. 15(a)]. In this work, spiking convolutional neural networks are used for feature extraction. To explain, consider the convolution kernel $W_{SC1}(i, j, k, 1)$. This kernel is used to find spikes at any location of the spiking input image. If there is a spike in the spiking image that matches up with the kernel, then this result will be a maximum (maximum correlation of the kernel with the image). The accumulated membrane potential

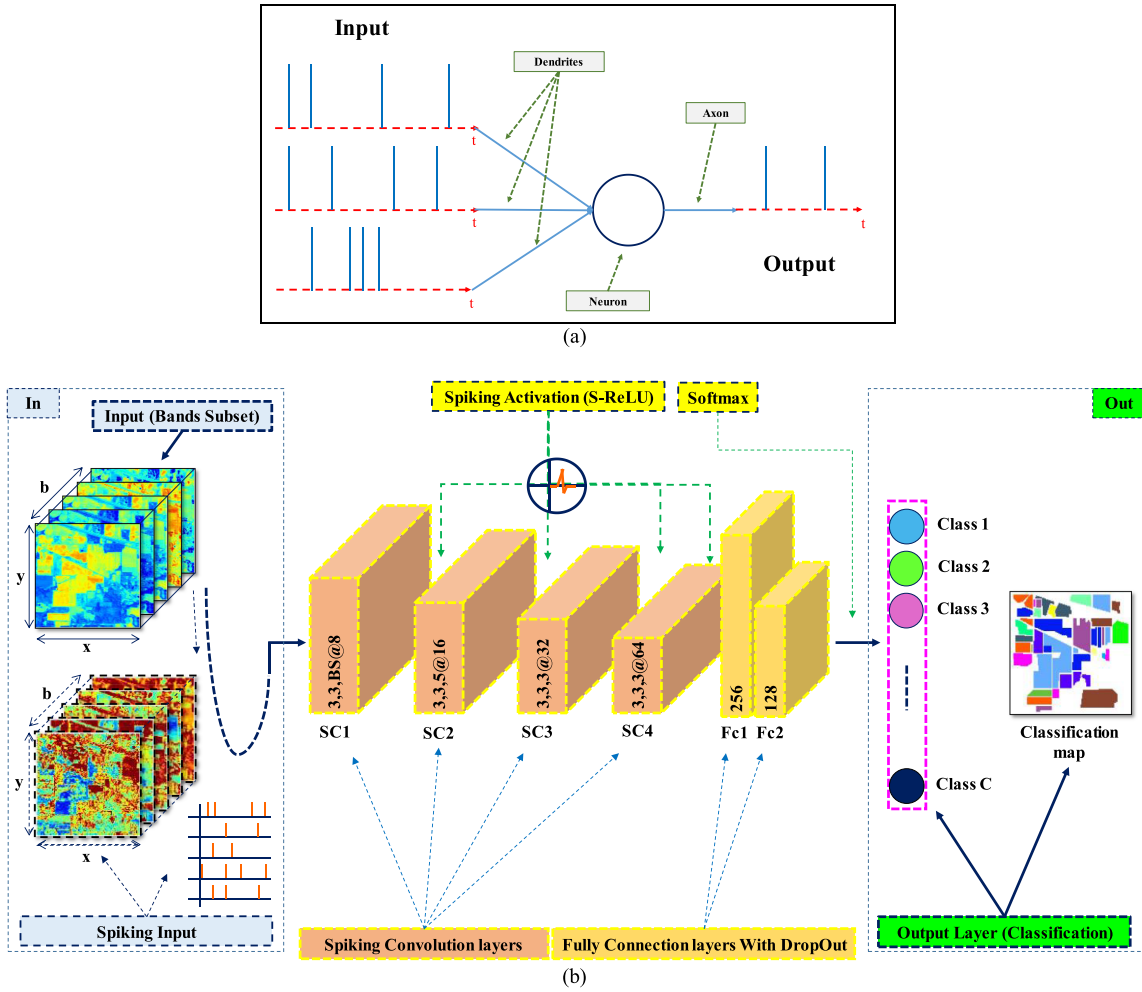


Fig. 15. (a) Structure spike neuron with input–output spike data. (b) Structure of the 3D-CNN model to SNN based on 3-D convolution layer and input data.

for the neuron at location (x, y, z) of map1 of the SC1 layer is given by

$$V_m(x, y, z, t, 1) = \sum_{\tau=0}^t \left(\sum_{i=1}^I \left(\sum_{j=1}^J \left(\sum_{k=1}^K (I(x + i, y + j, z + k, \tau) W_{SC1}(i, j, k, 1)) \right) \right) \right) \quad (2)$$

The neuron at (x, y, z) of map 1 of the SC1 layer then spikes at time t if

$$V_m^{(1)}(x, y, z, t, 1) \geq \gamma_{sc1} \quad (3)$$

where γ_{sc1} is the threshold. If the neuron at (x, y, z) in map 1 of SC1, then a vertical line of spikes has been detected in the spiking image centered at (x, y, z) . Similarly, feature maps will be generated in layers SC2 to 4 [92], [93], [94].

Fig. 15(b) shows the structure of the SNN based on convolutional layers. The structure of the 3D-CNN network model proposed in this article has been transformed into an SNN network without any fundamental changes. This conversion has been done using existing frameworks designed in Python.

Also, the spiking-ReLU activation function is used in the output section of each layer of the network. The input data have also been applied to the network in spiking form. To generate input data spiking form, the resulting image of the selected bands is passed through an on-center and an off-center difference of the Gaussian [DoG in (4)] convolution filter. The output of each of the two DoG filters is computed using the same mode of convolution. To generate input data spiking form, the resulting image of the selected bands is passed through an on-center and an off-center DoG convolution filter. The output of each of the two DoG filters is computed using the same mode of convolution

$$V_{\sigma_1, \sigma_2}(x, y) = \begin{cases} 1/(2\pi\sigma_1^2)e^{-(x^2+y^2)/(2\sigma_1^2)} \\ -1/(2\pi\sigma_2^2)e^{-(x^2+y^2)/(2\sigma_2^2)} \\ 0 \end{cases} \quad \text{for } a < x < b, a < y < b \quad \text{otherwise} \quad (4)$$

where σ_1 and σ_2 are for the on-center and off-center between a and b [95].

TABLE XIII
ACCURACY AND RUN-TIME 3D-CNN MODEL TO SNN MODEL

Dataset	Item	Model	Recommended sub-bands					
			5 BS	10 BS	15 BS	20 BS	25 BS	30 BS
IP	Run-Time	CNN	1368	2587	4395	5361	8591	10618
		SNN	638	1193	2218	2587	4089	4951
	Accuracy	CNN	0.9085	0.94003	0.95014	0.9505	0.9658	0.9661
		SNN	0.9015	0.9383	0.9457	0.9515	0.957	0.9622
	The Same as Run-Time CNN	SNN	0.9485	0.969	0.9847	0.9912	0.999	0.999
SA	Run-Time	CNN	3703	7729	11542	14246	19853	27439
		SNN	1824	3785	5693	7485	10266	13486
	Accuracy	CNN	0.93007	0.9573	0.9668	0.9728	0.9809	0.982
		SNN	0.9017	0.9236	0.9294	0.9541	0.9634	0.978
	The Same as Run-Time CNN	SNN	0.9418	0.9633	0.988	0.9958	0.999	0.999
PU	Run-Time	CNN	3842	7851	12642	15877	20438	29057
		SNN	1898	3999	6492	8047	11361	15702
	Accuracy	CNN	0.9807	0.9834	0.9927	0.9948	0.9949	0.9973
		SNN	0.9632	0.9688	0.9731	0.988	0.99	0.99
	The Same as Run-Time CNN	SNN	0.999	0.999	0.999	0.999	0.999	0.999
H	Run-Time	CNN	8251	15785	30215	35473	57163	72048
		SNN	3798	8294	15647	17653	27318	35473
	Accuracy	CNN	0.9206	0.9438	0.9573	0.9717	0.9749	0.9807
		SNN	0.9172	0.9257	0.9318	0.9667	0.969	0.984
	The Same as Run-Time CNN	SNN	0.9538	0.9673	0.9898	0.999	0.999	0.999

The classification accuracy with the new modified network has been tested only for all datasets. This advantage was investigated in the implementation of the proposed network. The results of checking the speed and execution time of the proposed network show that the speed and processing time compared to the 3D-CNN network with equal conditions show a reduction of more than 50%. Table XIII shows the classification accuracy values and the run-time (time complexity) of 3D-CNN and SNN models. Also, the best results are shown in bold.

This conversion has been done using existing frameworks designed in Python. To simulate the model and evaluate its performance, the Nengo library has been used in Python. Nengo is a Python library for building and simulating large-scale neural models. Nengo can create sophisticated spiking and nonspiking neural simulations with sensible defaults in a few lines of code. To study and access this library, visit <https://www.nengo.ai>. In the last stage of this experiment, we considered the conditions slightly different. For the same run-time in the 3D-CNN model, it was necessary to increase the number of epochs. The number of epochs for the same run-time and the SNN model is approximately 120 to 160. Also, the accuracy in the SNN model has increased to 4%. The results obtained in this case from the experiment were evaluated as suitable results compared to the 3D-CNN model. The results for equal execution time, accuracy for 5BS to 30BS mode, are obtained, as shown in Table XV. Reducing the parameters (due to changing the neurons of the network) is effective in improving the accuracy. For equal execution time, the number of network cycles is increased. According to these conditions, an increase in accuracy was expected. In

addition, energy consumption should be added to the advantages of this model of neural networks compared to the ANNs. Finally, Table XIV compares the accuracy value obtained from the classification of subbands in the methods evaluated in this article is displayed. The accuracy results of the proposed method have also been compared in three different modes (3D-CNN, 3D-CNN with attention layer, and 3D-CNN to SNN). As mentioned in the previous sections, the proposed method with the SNN neural network model has the highest classification accuracy compared to competing methods. According to the cases and advantages of SNN neural networks that were mentioned earlier, an increase in accuracy has been achieved. Fig. 16 also shows the accuracy comparison curve of the evaluated methods in this article.

G. Computational Complexity Analysis

After comparing the differences in accuracy, one of the most significant indices used to assess and evaluate the quality or quantity of approaches is the computational complexity index. The suggested technique includes the following mention of this index.

The complexity of the GA is affected by the population, the genetic operators and how they are implemented (which may have a considerable impact on total complexity), and obviously the fitness function. The GA has an $O(P * G * O(\text{Fitness}) * ((Pc * O(\text{crossover}) + (Pm * O(\text{mutation}))))$ complexity. The complexity depends on how many things there are, how many generations there are, and how long it takes to process each generation.

TABLE XIV
ACCURACY OF METHODS IN THE CLASSIFICATION OF SUBBANDS HSI DATASETS

Dataset	Methods	5 BS	10 BS	15 BS	20 BS	25 BS	30 BS
(IP)	SLN_BS	0.7061	0.7203	0.7331	0.7468	0.7481	0.75
	BS_UQ	0.85	0.908	0.937	0.9403	0.9491	0.9508
	LSI_BS	0.768	0.78	0.792	0.803	0.809	0.813
	3D-CNN	0.9085	0.94003	0.95014	0.9505	0.9658	0.9661
	3D-CNN_Att	0.9203	0.9518	0.9731	0.978	0.9827	0.9812
	3D-CNN_SNN	0.9485	0.969	0.9847	0.9912	0.999	0.999
Salinas (SA)	SLN_BS	0.905	0.9086	0.9103	0.9119	0.9106	0.9109
	BS_UQ	0.908	0.9164	0.937	0.9481	0.9503	0.972
	LSI_BS	0.879	0.884	0.89	0.8955	0.896	0.8967
	3D-CNN	0.93007	0.9573	0.9668	0.9728	0.9809	0.982
	3D-CNN_Att	0.938	0.9651	0.97	0.988	0.99	0.994
	3D-CNN_SNN	0.9418	0.9633	0.988	0.9958	0.999	0.999
Pavia University (PU)	SLN_BS	0.8895	0.9016	0.9207	0.9461	0.9482	0.9519
	BS_UQ	0.906	0.92	0.9125	0.9325	0.952	0.97
	LSI_BS	0.773	0.8	0.806	0.809	0.813	0.818
	3D-CNN	0.9807	0.9834	0.9927	0.9948	0.9949	0.9973
	3D-CNN_Att	0.998	0.998	0.998	0.999	0.999	0.999
	3D-CNN_SNN	0.999	0.999	0.999	0.999	0.999	0.999
Houston (H)	SLN_BS	0.892	0.9104	0.9341	0.9372	0.9418	0.9437
	BS_UQ	0.87	0.9120	0.939	0.945	0.952	0.9583
	LSI_BS	0.714	0.746	0.735	0.751	0.75	0.762
	3D-CNN	0.9206	0.9438	0.9573	0.9717	0.9749	0.9807
	3D-CNN_Att	0.9528	0.9614	0.9704	0.9882	0.99	0.9913
	3D-CNN_SNN	0.9538	0.9673	0.9898	0.999	0.999	0.999

Additionally, the complexity is transformed to O (O (Fitness) * (O (mutation) + O (crossover))), and P , G , P_c , and P_m are constants. The large o is $O(1)$ since it takes a given amount of time and the number of generations and population size are constant. This holds true for mutation functions, crossover functions, and fitness functions as long as they take a known amount of time. Considering that the convolutional neural network is used as the fitness function in GA, the complexity of the proposed method depends a lot on the computational complexity of the 3D-CNN network. The number of operations in each layer should be calculated initially since the complexity of CNN relies on various levels. Then, all the complexity of CNN relies on various levels. Then, all of these operations must be added, and time complexity must be expressed as a function of the input (and probably the number of layers). The complexity of a 1-D convolutional layer $O(k*n*d)$ and a 1-D convolution is the sum of the rowwise dot products of a filter $W \in \mathbb{R}^{k \times d}$ with a region matrix $A \in \mathbb{R}^{k \times d}$, where k is the length of the filter and d is the depth dimension (e.g., dimensionality of word embedding space), and finally, at the layer level, we apply the filter over the input $n-k+1$ times (where n is the length of the input), let us say n times since $n \gg k$. This gives us a final complexity of $O(n*k*d)$. However, the computational complexity of 3-D convolution for images with $N * M * K$ dimensions and $N * M * K$ filter sizes is equivalent to $O(NMKnmk)$. Therefore,

TABLE XV
COMPLEXITY OF THE ALL METHODS

Methods	Complexity
SLN_BS	$\sim O(n * d * k) \text{ to } O(n^2)$
BS_UQ	$\sim O(m^2 * n) \text{ to } O(m^3 * n)$
LSI_BS	$\sim O(m^2 * n) \text{ to } O(m^3 * n)$
CNNeGA	$\sim O(m^4 * n)$ $(O(knmKNM * \text{Num. Layers}) * (O(\text{mutation}) + O(\text{crossover})))$
3D-CNN_Att	$\sim O(L * m^4 * n)$, $L = \text{Number of added layers}$
3D-CNN_SNN	$\sim O(1/2 * m^4 * n)$ <i>Reduced Run-Time compared to 3D-CNN</i>

the computational complexity of 3-D convolution for pictures with $N * M * K$ dimensions and $N * M * K$ filter sizes is equivalent to $O(NMKnmk)$. As a result, the complexity in the 3D-CNN network, which has four 3-D CNN layers, will be $O(knmKNM^{\text{Num. Layers}})$. Regarding the proposed method in this article, the complexity of calculations will be equal to

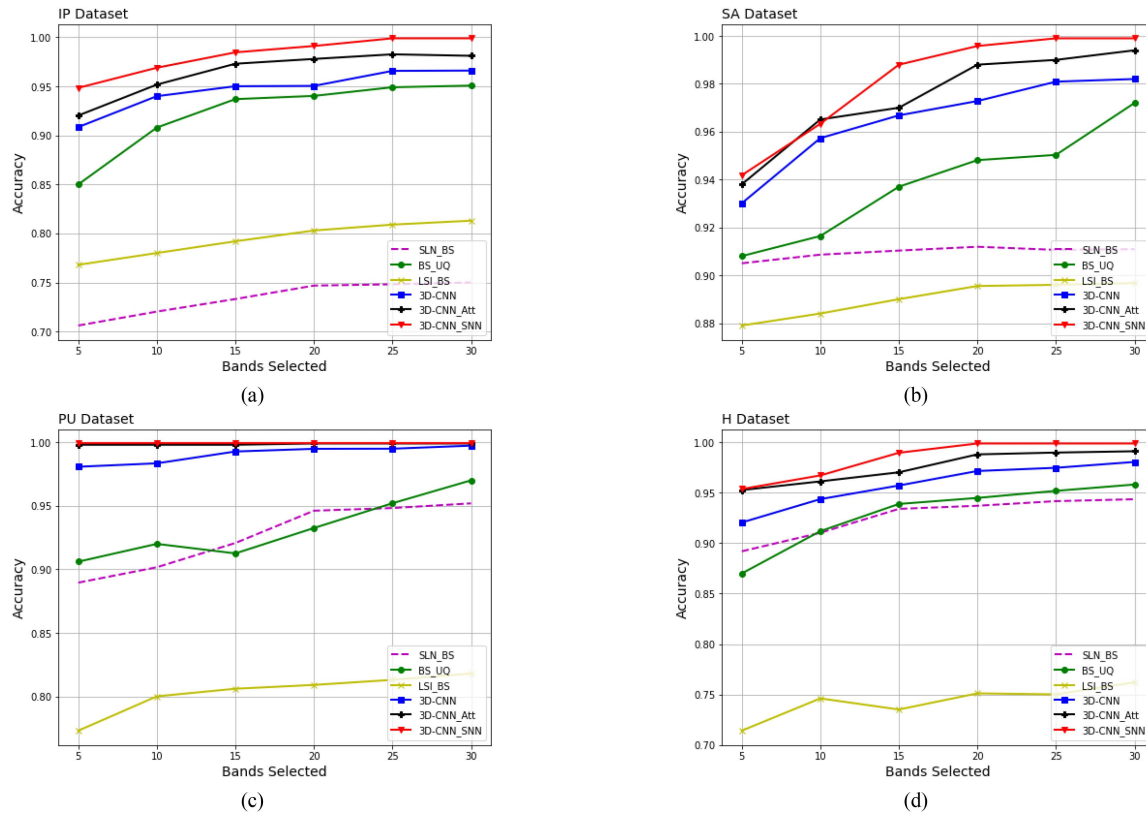


Fig. 16. Comparison curve of the accuracies of classification obtained in the proposed method with the methods presented in the different datasets, (a) IP, (b) Salinas, (c) Pavia University, and (d) Houston dataset.

the product of the complexity of the 3D-CNN network and the complexity of the GA $O(O(\text{knmKNM}^{\text{Num. Layers}}) * (O(\text{mutation}) + O(\text{crossover})))$. In Table XV, the complexity of all methods (BS_UQ, LSI_BS, SLN_BS, CNNeGA) and also the complexity of the proposed method with 3D-CNN with Att. Layers and SNN are displayed. The complexity of the proposed method is also evident as a result of the changed structure of the 3D-CNN model (adding path and attention layers). Therefore, the complexity of 3D-CNN_Att will be the number of additional layers (L) multiplied by the complexity of CNNeGA. Although 3D-CNN_SNN has the same structure as 3D-CNN, it is less complex due to the change in the nature of the input data and fewer parameters. The complexity of 3D-CNN_SNN based on the run-time option is almost half of CNNeGA.

V. CONCLUSION

BS is an effective way to reduce the size of hyperspectral data and to overcome the curse of dimensionality problem in ground object classification. This study proposes a supervised BS framework based on a combination of convolutional neural networks and the GA algorithm in this context. Convolutional networks and metaheuristic search techniques are used in the fundamental concept and proposed architecture of the BS for HSI. A subset of bands that accurately reflect the original bands and have little redundancy can be chosen using this framework. The CNNeGA approach was fairly resilient to different noise

bands. A group of bands corresponding to a higher level of classification accuracy was selected. The bands with the highest correlation to the selected bands were automatically disregarded in the suggested method, which is an iterative procedure. The suggested approach could be thought of as a class-based approach in general. With this strategy, a BS criterion that suits the requirements of each class was possible. In other techniques, bands are chosen depending on the statistical characteristics of the dataset. The proposed method in this work has the advantage of being classification-based and uses a linear classification to rank and choose bands. The proposed method (CNNeGA) shows between 6% and 21% performance improvements compared to its competitors in the experimental data reviewed in this study. According to the experimental data, the band subset chosen by the CNNeGA approach was more successful at classification and had lower performance and correlation than the band subsets chosen by other BS methods. Testing on different types of datasets has shown that the proposed method is more stable than competing methods in terms of scale, number of samples, and multiplicity of dataset classes. Finally, to improve the performance of the proposed neural network model (3D-CNN) integrated with GA, the number of attention layers was increased, and the structure of the model was changed to SNNs. The accuracy of classification has increased by approximately 1% to 3% by changing the network structure and model. In the computational complexity index, the complexity time has been reduced by more than 50%. The increase in classification

accuracy and processing speed (reduction of computational complexity), especially in changing the model to spike networks, was significant and promising for future work. According to the obtained results, we intend to investigate the application of the SNN model for the problem of BS and classification of HSIs in an unsupervised manner.

REFERENCES

- [1] S. S. Wahla, J. H. Kazmi, A. Sharifi, S. A. Shirazi, A. Tariq, and H. Joyell Smith, "Assessing spatio-temporal mapping and monitoring of climatic variability using SPEI and RF machine learning models," *Geocarto Int.*, vol. 38, 2022, Art. no. 21, doi: [10.1080/10106049.2022.2093411](https://doi.org/10.1080/10106049.2022.2093411).
- [2] A. Sharifi, H. Mahdipour, E. Moradi, and A. Tariq, "Agricultural field extraction with deep learning algorithm and satellite imagery," *J. Indian Soc. Remote Sens.*, vol. 50, pp. 417–423, 2022, doi: [10.1007/s12524-021-01475-7](https://doi.org/10.1007/s12524-021-01475-7).
- [3] A. Tariq, S. Siddiqui, A. Sharifi, and S. H. I. A. Shah, "Impact of spatio-temporal land surface temperature on cropping pattern and land use and land cover changes using satellite imagery, Hafizabad District, Punjab, Province of Pakistan," *Arabian J. Geosci.*, vol. 15, 2022, Art. no. 1045, doi: [10.1007/s12517-022-10238-8](https://doi.org/10.1007/s12517-022-10238-8).
- [4] P. K. Varshney and M. K. Arora, *Advanced Image Processing Techniques for Remotely Sensed Hyperspectral Data*. Berlin, Germany: Springer, 2004.
- [5] S. Jalayer, A. Sharifi, D. Abbasi-Moghadam, A. Tariq, and S. Qin, "Modeling and predicting land use land cover spatiotemporal changes: A case study in Chalus Watershed, Iran," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 5496–5513, 2022, doi: [10.1109/JS-TARS.2022.3189528](https://doi.org/10.1109/JS-TARS.2022.3189528).
- [6] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.
- [7] X. Miao, P. Gong, S. Swope, R. Pu, R. Carruthers, and G. L. Anderson, "Detection of yellow starthistle through band selection and feature extraction from hyperspectral imagery," *Photogrammetric Eng. Remote Sens.*, vol. 73, no. 9, pp. 1005–1015, 2007.
- [8] S. M. M. Nejad, D. Abbasi-Moghadam, A. Sharifi, N. Farmonov, K. Amankulova, and M. Laszlo, "Multi-spectral crop yield prediction using 3D-convolutional neural networks and attention convolutional LSTM approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 254–266, 2023, doi: [10.1109/JSTARS.2022.3223423](https://doi.org/10.1109/JSTARS.2022.3223423).
- [9] I. Dópidó, A. Villa, A. Plaza, and P. Gamba, "A quantitative and comparative assessment of unmixing-based feature extraction techniques for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 421–435, Apr. 2012.
- [10] A. Martínez-UsoMartínez-Uso, F. Pla, J. M. Sotoca, and P. García-Sevilla, "Clustering-based hyperspectral band selection using information measures," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4158–4171, Dec. 2007.
- [11] H. Yang, Q. Du, and G. Chen, "Particle swarm optimization-based hyperspectral dimensionality reduction for urban land cover classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 544–554, Apr. 2012.
- [12] S. Chen, L. Liu, L. Sui, and X. Liu, "Improving GPP estimates by partitioning green APAR from total APAR in two deciduous forest sites," *J. Forestry Res.*, 2022, doi: [10.1007/s11676-022-01546-6](https://doi.org/10.1007/s11676-022-01546-6).
- [13] C. Miao, P. Li, Y. Huang, Y. Sun, W. Chen, and S. Yu, "Coupling outdoor air quality with thermal comfort in the presence of street trees: A pilot investigation in Shenyang, Northeast China," *J. Forestry Res.*, 2022, doi: [10.1007/s11676-022-01497-y](https://doi.org/10.1007/s11676-022-01497-y).
- [14] M. Zhang, J. Ma, and M. Gong, "Unsupervised hyperspectral band selection by fuzzy clustering with particle swarm optimization," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 773–777, May 2017.
- [15] S. S. Sawant and M. Prabukumar, "A survey on band selection techniques for hyperspectral image classification," *J. Spectral Imag.*, vol. 9, 2020, Art. no. a5.
- [16] A. Sharifi, "Development of a method for flood detection based on Sentinel-1 images and classifier algorithms," *Water Environ. J.*, vol. 35, pp. 924–929, 2021, doi: [10.1111/wej.12681](https://doi.org/10.1111/wej.12681).
- [17] W. Zhang, X. Li, and L. Zhao, "Discovering the representative subset with low redundancy for hyperspectral feature selection," *Remote Sens.*, vol. 11, 2019, Art. no. 1341.
- [18] C.-I. Chang, Q. Du, T.-L. Sun, and M. L. Althouse, "A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 6, pp. 2631–2641, Nov. 1999.
- [19] P. Bajcsy, "Methodology for hyperspectral band selection," *Photogrammetric Eng. Remote Sens. J.*, vol. 70, no. 7, pp. 793–802, Jul. 2004.
- [20] C.-I. Chang, Q. Du, T.-L. Sun, and M. L. G. Althouse, "A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 6, pp. 2631–2641, Nov. 1999.
- [21] P. Bajcsy and P. Groves, "Methodology for hyperspectral band selection," *Photogrammetric Eng. Remote Sens.*, vol. 70, no. 7, pp. 793–802, 2004.
- [22] C.-I. Chang and S. Wang, "Constrained band selection for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1575–1585, Jun. 2006.
- [23] B. Somers and G. P. Asner, "Multi-temporal hyperspectral mixture analysis and feature selection for invasive species mapping in rainforests," *Remote Sens. Environ.*, vol. 136, pp. 14–27, 2013.
- [24] C.-I. Chang and K. H. Liu, "Progressive band selection of spectral unmixing for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 4, pp. 2002–2017, Apr. 2014.
- [25] S. Jia, G. Tang, J. Zhu, and Q. Li, "A novel ranking-based clustering approach for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 88–102, Jan. 2016.
- [26] S. Feng, Y. Itoh, M. Parente, and M. F. Duarte, "Hyperspectral band selection from statistical wavelet models," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 4, pp. 2111–2123, Apr. 2017.
- [27] B. Guo, S. R. Gunn, R. I. Damper, and J. D. B. Nelson, "Band selection for hyperspectral image classification using mutual information," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 4, pp. 522–526, Oct. 2006.
- [28] D. Varade, A. K. Maurya, and O. Dikshit, "Unsupervised band selection of hyperspectral data based on mutual information derived from weighted cluster entropy for snow classification," *Geocarto Int.*, vol. 36, pp. 1709–1731, 2021.
- [29] C. Cariou, K. Chehdi, and S. Le Moan, "BandClust: An unsupervised band reduction method for hyperspectral remote sensing," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 3, pp. 565–569, May 2011.
- [30] H. Su, H. Yang, Q. Du, and Y. Sheng, "Semisupervised band clustering for dimensionality reduction of hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 6, pp. 1135–1139, Nov. 2011.
- [31] X. Cao, B. Wu, D. Tao, and L. Jiao, "Automatic band selection using spatial-structure information and classifier-based clustering," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 9, pp. 4352–4360, Sep. 2016.
- [32] R. Yang, L. Su, X. Zhao, H. Wan, and J. Sun, "Representative band selection for hyperspectral image classification," *J. Vis. Commun. Image Representation*, vol. 48, pp. 396–403, 2017.
- [33] M. Zeng, Y. Cai, Z. Cai, X. Liu, P. Hu, and J. Ku, "Unsupervised hyperspectral image band selection based on deep subspace clustering," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 12, pp. 1889–1893, Dec. 2019.
- [34] S. Bhattacharya, S. Das, and A. Routray, "Graph manifold clustering-based band selection for hyperspectral face recognition," in *Proc. 26th Eur. Signal Process. Conf.*, 2018, pp. 1990–1994.
- [35] Y. Liu, H. Xie, L. Wang, and K. Tan, "Hyperspectral band selection based on a variable precision neighborhood rough set," *Appl. Opt.*, vol. 55, no. 3, 2016, Art. no. 462.
- [36] X. Cao, C. Wei, J. Han, and L. Jiao, "Hyperspectral band selection using improved classification map," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 2147–2151, Nov. 2017.
- [37] F. Xie, F. Li, C. Lei, and L. Ke, "Representative band selection for hyperspectral image classification," *ISPRS Int. J. Geo-Inf.*, vol. 7, no. 9, 2018, Art. no. 338.
- [38] P. Ghamisi and J. A. Benediktsson, "Feature selection based on hybridization of genetic algorithm and particle swarm optimization," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 2, pp. 309–313, Feb. 2015.
- [39] H. Yang, Q. Du, and G. Chen, "Particle swarm optimization-based hyperspectral dimensionality reduction for urban land cover classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 544–554, Apr. 2012.
- [40] H. Su, B. Yong, and Q. Du, "Hyperspectral band selection using improved firefly algorithm," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 1, pp. 68–72, Jan. 2016.

- [41] M. Habermann, V. Frémont, and E. H. Shiguemori, "Supervised band selection in hyperspectral images using single-layer neural networks," *Int. J. Remote Sens.*, vol. 40, no. 10, pp. 3900–3926, 2018, doi: [10.1080/01431161.2018.1553322](https://doi.org/10.1080/01431161.2018.1553322).
- [42] R. Aghaee, M. Momeni, and P. Moallem, "Semisupervised band selection from hyperspectral images using Levy flight-based genetic algorithm," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6006805, doi: [10.1109/LGRS.2022.3147272](https://doi.org/10.1109/LGRS.2022.3147272).
- [43] Z. Ye, W. Cai, S. Liu, K. Liu, M. Wang, and W. Zhou, "A band selection approach for hyperspectral image based on a modified hybrid rice optimization algorithm," *Symmetry*, vol. 14, 2022, Art. no. 1293, doi: [10.3390/sym14071293](https://doi.org/10.3390/sym14071293).
- [44] Y. Yuan, G. Zhu, and Q. Wang, "Hyperspectral band selection by multitasking sparsity pursuit," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 631–644, Feb. 2015.
- [45] Z. Dou, K. Gao, X. Zhang, H. Wang, and L. Han, "Band selection of hyperspectral images using attention-based autoencoders," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 1, pp. 147–151, Jan. 2021.
- [46] J. Shi, X. Zhang, X. Liu, Y. Lei, and G. Jeon, "Multicriteria semi-supervised hyperspectral band selection based on evolutionary multitask optimization," *Knowl.-Based Syst.*, vol. 240, 2022, Art. no. 107934, doi: [10.1016/j.knsys.2021.107934](https://doi.org/10.1016/j.knsys.2021.107934).
- [47] A. Paul and N. Chaki, "Supervised data-driven approach for hyperspectral band selection using quantization," *Geocarto Int.*, vol. 37, pp. 2312–2322, 2022, doi: [10.1080/10106049.2020.1822929](https://doi.org/10.1080/10106049.2020.1822929).
- [48] X. Cao, T. Xiong, and L. Jiao, "Supervised band selection using local spatial information for hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 329–333, Mar. 2016, doi: [10.1109/LGRS.2015.2511186](https://doi.org/10.1109/LGRS.2015.2511186).
- [49] M. You, X. Meng, Y. Wang, and H. Jin, "Hyperspectral band selection via band grouping and adaptive multi-graph constraint," *Remote Sens.*, vol. 14, 2022, Art. no. 4379, doi: [10.3390/rs14174379](https://doi.org/10.3390/rs14174379).
- [50] Y. Qian, M. Ye, and J. Zhou, "Hyperspectral image classification based on structured sparse logistic regression and three-dimensional wavelet texture features," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 4, pp. 2276–2291, Apr. 2013.
- [51] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2020.
- [52] Y. Li and L. He, "An improved hybrid CNN for hyperspectral image classification," in *Proc. 11th Int. Conf. Graph. Image Process.*, 2020, pp. 485–490, doi: [10.1117/12.2557384](https://doi.org/10.1117/12.2557384).
- [53] B. Fang, Y. Bai, and Y. Li, "Combining spectral unmixing and 3D/2D dense networks with the early-exiting strategy for hyperspectral image classification," *Remote Sens.*, vol. 12, 2020, Art. no. 779.
- [54] M. He, B. Li, and H. Chen, "Multi-scale 3D deep convolutional neural network for hyperspectral image classification," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 3904–3908.
- [55] L. Huang and Y. Chen, "Dual-path siamese CNN for hyperspectral image classification with limited training samples," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 3, pp. 518–522, Mar. 2021.
- [56] A. Sharifi, "Flood mapping using relevance vector machine and SAR data: A case study from Aqqa, Iran," *J. Indian Soc. Remote Sens.*, vol. 48, pp. 1289–1296, 2020, doi: [10.1007/s12524-020-01155-y](https://doi.org/10.1007/s12524-020-01155-y).
- [57] S. Mittal and Vibhu, "A survey of accelerator architectures for 3D convolution neural networks," *J. Syst. Architecture*, vol. 115, 2021, Art. no. 102041, doi: [10.1016/j.sysarc.2021.102041](https://doi.org/10.1016/j.sysarc.2021.102041).
- [58] K. Gavahi, P. Abbaszadeh, and H. Moradkhani, "DeepYield: A combined convolutional neural network with long short-term memory for crop yield forecasting," *Expert Syst. Appl.*, vol. 184, 2021, Art. no. 115511.
- [59] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 807–814.
- [60] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, Aug. 2002, doi: [10.1162/08997660276012801](https://doi.org/10.1162/08997660276012801).
- [61] S. Ghaderizadeh, D. Abbasi-Moghadam, A. Sharifi, N. Zhao, and A. Tariq, "Hyperspectral image classification using a hybrid 3D–2D convolutional neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 7570–7588, 2021.
- [62] S. N. Sivanandam and S. N. Deepa, *Introduction to Genetic Algorithms*. Berlin, Germany: Springer, 2008, doi: [10.1007/978-3-540-73190-0](https://doi.org/10.1007/978-3-540-73190-0).
- [63] Z. Michalewicz and M. Schoenauer, "Evolutionary algorithms for constrained parameter optimization problems," *Evol. Comput.*, vol. 4, no. 1, pp. 1–32, 1996.
- [64] G. Kandavanam, D. Botvich, S. Balasubramaniam, and B. Jennings, "A hybrid genetic algorithm/variable neighborhood search approach to maximizing residual bandwidth of links for route planning," in *Proc. Int. Conf. Artif. Evol.*, 2010, pp. 49–60.
- [65] S. S. Chouhan, A. Kaul, and U. P. Singh, "Soft computing approaches for image segmentation: A survey," *Multimed Tools Appl.*, vol. 77, no. 21, pp. 28483–28537, 2018.
- [66] M. Kaur and V. Kumar, "Parallel non-dominated sorting genetic algorithm-II-based image encryption technique," *Imag. Sci. J.*, vol. 66, no. 8, pp. 453–462, 2018.
- [67] A. R. Kavitha and C. Chellamuthu, "Brain tumor segmentation from MRI image using genetic algorithm with fuzzy initialization and seeded modified region growing (GFSMRG) method," *Imag. Sci. J.*, vol. 64, no. 5, pp. 285–297, 2016.
- [68] E. Y. Kim and K. Jung, "Genetic algorithms for video segmentation," *Pattern Recognit.*, vol. 38, no. 1, pp. 59–73, 2005.
- [69] A. Peerlinck, J. Sheppard, J. Pastorino, and B. Maxwell, "Optimal design of experiments for precision agriculture using a genetic algorithm," in *Proc. IEEE Congr. Evol. Comput.*, 2019, pp. 1838–1845.
- [70] C. Lee, "A review of applications of genetic algorithms in operations management," *Eng. Appl. Artif. Intell.*, vol. 76, pp. 1–12, 2018.
- [71] G. Sermpinis, C. Stasinakis, K. Theofilatos, and A. Karathanasopoulos, "Modeling, forecasting and trading the EUR exchange rates with hybrid rolling genetic algorithms—Support vector regression forecast combinations," *Eur. J. Oper. Res.*, vol. 247, no. 3, pp. 831–846, 2015.
- [72] Z. Michalewicz, *Genetic Algorithms, Data Structures = Evolution Programs*. Berlin, Germany: Springer, 1996.
- [73] M. Mitchell, *Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1996.
- [74] D. Whitley, "A genetic algorithm tutorial," *Statist. Comput.*, vol. 4, no. 2, pp. 65–85, 1994.
- [75] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on the genetic algorithm: Past, present, and future," *Multimed Tools Appl.*, vol. 80, no. 5, pp. 8091–8126, 2021.
- [76] Y. Yuan, G. Zhu, and Q. Wang, "Hyperspectral band selection by multitask sparsity pursuit," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 631–644, Feb. 2015, doi: [10.1109/TGRS.2014.2326655](https://doi.org/10.1109/TGRS.2014.2326655).
- [77] P. S. Singh and S. Karthikeyan, "Enhanced classification of remotely sensed hyperspectral images through efficient band selection using autoencoders and genetic algorithm," *Neural Comput. Appl.*, vol. 34, pp. 21539–21550, 2022, doi: [10.1007/s00521-021-06121-4](https://doi.org/10.1007/s00521-021-06121-4).
- [78] J. Shi, X. Zhang, X. Liu, and Y. Lei, "Multicriteria semi-supervised hyperspectral band selection based on evolutionary multitask optimization," *Knowl.-Based Syst.*, vol. 240, Mar. 2022, Art. no. 107934, doi: [10.1016/j.knsys.2021.107934](https://doi.org/10.1016/j.knsys.2021.107934).
- [79] X. Ke and Y. Zhang, "Fine-grained vehicle type detection and recognition based on dense attention network," *Neuro-Comput.*, vol. 399, pp. 247–257, 2020, doi: [10.1016/j.neucom.2020.02.101](https://doi.org/10.1016/j.neucom.2020.02.101).
- [80] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7794–7803, doi: [10.1109/CVPR.2018.00813](https://doi.org/10.1109/CVPR.2018.00813).
- [81] A. Vaswani et al., "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
- [82] S. Woo, J. Park, and J. Lee, "CBAM: Convolutional block attention module," in *Computer Vision*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Berlin, Germany: Springer, Oct. 2018, pp. 3–19.
- [83] M. Zhu, L. Jiao, F. Liu, S. Yang, and J. Wang, "Residual spectral-spatial attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 449–462, Jan. 2021.
- [84] X. Zhang, Y. Sun, K. Jiang, C. Li, L. Jiao, and H. Zhou, "Spatial sequential recurrent neural network for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 11, pp. 4141–4155, Nov. 2018.
- [85] Y. Peng, Y. Zhang, B. Tu, Q. Li, and W. Li, "Spatial-spectral transformer with cross-attention for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5537415, doi: [10.1109/TGRS.2022.3203476](https://doi.org/10.1109/TGRS.2022.3203476).
- [86] S. Ghaderizadeh, D. Abbasi-Moghadam, A. Sharifi, A. Tariq, and S. Qin, "Multiscale dual-branch residual spectral-spatial network with attention for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 5455–5467, 2022, doi: [10.1109/JS-TARS.2022.3188732](https://doi.org/10.1109/JS-TARS.2022.3188732).

- [87] L. Dang, L. Weng, W. Dong, S. Li, and Y. Hou, "Spectral-spatial attention transformer with dense connection for hyperspectral image classification," *Hindawi Comput. Intell. Neurosci.*, 2022, Art. no. 7071485, doi: [10.1155/2022/7071485](https://doi.org/10.1155/2022/7071485).
- [88] R. Vaïla, J. Chiasson, and V. Saxena, "Feature extraction using spiking convolutional neural networks," in *Proc. Int. Conf. Neuromorphic Syst.*, Jul. 2019, Art. no. 14, doi: [10.1145/3354265.3354279](https://doi.org/10.1145/3354265.3354279).
- [89] A. Kahana, Q. Zhang, L. Gleyzer, and G. E. Karniadakis, "Spiking neural operators for scientific machine learning," 2022, *arXiv.2205.10130*.
- [90] A. Sharifi, J. Amini, and R. Tateishi, "Estimation of forest biomass using multivariate relevance vector regression," *Photogramm. Eng. Remote Sens.*, vol. 82, no. 1, pp. 41–49, 2016, doi: [10.14358/PERS.83.1.41](https://doi.org/10.14358/PERS.83.1.41).
- [91] J. Büchel, D. Zendrikov, S. Solinas, G. Indiveri, and D. R. Muir, "Supervised training of spiking neural networks for robust deployment on mixed-signal neuromorphic processors," *Sci. Rep.*, vol. 11, no. 1, 2021, doi: [10.1038/s41598-021-02779-x](https://doi.org/10.1038/s41598-021-02779-x).
- [92] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida, "Deep learning in spiking neural networks," *Neural Netw.*, vol. 111, pp. 47–63, 2019, doi: [10.1016/j.neunet.2018.12.002](https://doi.org/10.1016/j.neunet.2018.12.002).
- [93] S. R. Shirsavar, A.-H. Vahabie, and M.-R. A. Dehaqani, "Models developed for spiking neural networks," 2022, *arXiv.2212.04377*.
- [94] L. Paulun, A. Wendt, and N. Kasabov, "A retinotopic spiking neural network system for accurate recognition of moving objects using neucube and dynamic vision sensors," *Front. Comput. Neurosci.*, vol. 12, 2018, doi: [10.3389/fncom.2018.00042](https://doi.org/10.3389/fncom.2018.00042).
- [95] G. Orchard, A. Jayawant, G. K. Cohen, and N. Thakor, "Converting static image datasets to spiking neuromorphic datasets using saccades," *Front. Neurosci.*, vol. 9, 2015, Art. no. 437.



Alireza Sharifi was born in Tehran, Iran, in 1981. He received the M.Sc. and Ph.D. degrees in remote sensing engineering from the University of Tehran, Tehran, Iran, in 2008 and 2015, respectively.

He is currently an Assistant Professor of remote sensing with the Faculty of Civil Engineering, Shahid Rajaei Teacher Training University, Tehran, Iran. In particular, he is involved in GEOAI program for food security and environmental monitoring.



Aqil Tariq received the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan, China, in 2021.

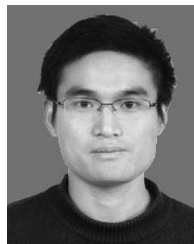
He is currently with the Department of Wildlife, Fisheries and Aquaculture, Mississippi State University, Mississippi State, MS, USA. His research interests include photogrammetry, 3-D geo-information, urban analytics, spatial analysis to examine land use/land cover, geospatial data science, agriculture monitoring, natural hazards (forest fire, landslide, droughts, and flood) forest monitoring, forest cover dynamics, spatial statistics, multicriteria algorithms, ecosystem sustainability, hazards risk reduction, statistical analysis, and modeling using Python, R, and MATLAB.



Mohammad Esmaceli was born in Kerman, Iran. He received the M.Sc. degree in electrical and communication engineering in 2012 from the Department of Technical and Engineering, Shahid Bahonar University of Kerman, Kerman, Iran, where he is currently working toward the Ph.D. degree in electrical and communication engineering.

He is currently researching deep learning methods for approaches to processing hyperspectral and multispectral images. His research interests include image processing, deep learning, hyperspectral remote sensing,

and its applications on hyper and multispectral images.



Qingting Li received the Ph.D. degree in remote sensing from the Institute of Remote Sensing Applications, Chinese Academy of Sciences (CAS), Beijing, China, in 2009.

He is currently an Associate Professor with the Aerospace Information Research Institute, CAS, Beijing, China. His research interests include hyperspectral remote sensing, thermal infrared remote sensing, time-series analysis, and monitoring of natural resources and eco-environment with earth observation data.



Dariush Abbasi-Moghadam received the B.S. degree in electrical engineering from the Shahid Bahonar University of Kerman (SBUK), Kerman, Iran, in 1998, and the M.S. and Ph.D. degrees in electrical engineering from the Iran University of Science and Technology, Tehran, Iran, in 2001 and 2011, respectively.

From 2001 to 2003, he was primary with the Advanced Electronic Research Center—Iran and worked on the design and analysis of satellite communication systems. In September 2004, he joined Iranian

Telecommunications Company, Tehran, Iran, as a Research Engineer. He is currently with the Department of Electrical Engineering, SBUK, as an Associate Professor. His research interests include wireless communications, satellite communication systems, remote sensing, and signal processing.