Turkish Journal of Electrical Engineering and Computer Sciences

Volume 30 | Number 6 Article 9

1-1-2022

Analysis of patch and sample size effects for 2D-3D CNN models using multiplatform dataset: hyperspectral image classification of ROSIS and Jilin-1 GP01 imagery

TAŞKIN KAVZOĞLU

ELİF ÖZLEM YILMAZ

Follow this and additional works at: https://journals.tubitak.gov.tr/elektrik

 Part of the Computer Engineering Commons, Computer Sciences Commons, and the Electrical and Computer Engineering Commons

Recommended Citation
KAVZOĞLU, TAŞKIN and YILMAZ, ELİF ÖZLEM (2022) "Analysis of patch and sample size effects for 2D-3D CNN models using multiplatform dataset: hyperspectral image classification of ROSIS and Jilin-1 GP01 imagery," *Turkish Journal of Electrical Engineering and Computer Sciences*: Vol. 30: No. 6, Article 9.
https://doi.org/10.55730/1300-0632.3929
Available at: https://journals.tubitak.gov.tr/elektrik/vol30/iss6/9

This Article is brought to you for free and open access by TÜBİTAK Academic Journals. It has been accepted for inclusion in Turkish Journal of Electrical Engineering and Computer Sciences by an authorized editor of TÜBİTAK Academic Journals. For more information, please contact academic.publications@tubitak.gov.tr.

Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2022) 30: 2124 – 2144
© TÜBİTAK
doi:10.55730/1300-0632.3929

Analysis of patch and sample size effects for 2D-3D CNN models using multiplatform dataset: hyperspectral image classification of ROSIS and Jilin-1 GP01 imagery

Taskın KAVZOGLU*, **Elif Ozlem YILMAZ**
Department of Geomatics Engineering, Faculty of Engineering, Gebze Technical University, Kocaeli, Turkey

Received: 25.03.2022 • **Accepted/Published Online:** 07.07.2022 • **Final Version:** 28.09.2022

Abstract: Modern hyperspectral sensors provide a huge volume of data at spectral and spatial domains with high redundancy, which requires robust methods for analysis. In this study, 2D and 3D CNN models were applied to hyperspectral image datasets (ROSIS and Jilin-1 GP01) using varying patch and sample sizes to determine their combined impacts on the performance of deep learning models. Differences in classification performances in relation to particle and sample sizes were statistically analysed using McNemar's test. According to the findings, raising the patch and sample size enhances the performance of the 2D/3D CNN model and produces more accurate results in the classification of hyperspectral imagery. To be more specific, the thematic maps produced with 400 and 600 samples resulted in a notable increase in overall accuracies (approximately 10%) compared to 50 and 100 samples per class. Statistical test results confirmed the importance of patch size selection for the case of limited training data and the maximum number of samples required for deep learning. It could be also mentioned that an appropriate patch and sample size should be selected considering the characteristics of the dataset and the type of deep learning algorithms.

Key words: Classification, convolutional neural network, deep learning, hyperspectral imagery, Jilin-1 GP01, patch size, sample size

1. Introduction

Over the years, the number of sensors monitoring the land surface has boosted; as a result, the amount of data collected by these technologies has exponentially increased with higher spatial and spectral resolution [1]. In earlier applications, only spectral information was utilized to extract information or features from satellite images using traditional pixel-based methods [2, 3]. These approaches were not effective due to the difficulties inherent in satellite images, including spectral, textural, and geometrical features [4]. Deep learning approach was lately introduced to overcome the difficulties by producing higher accuracy, particularly for image processing and classification problems [6, 7]. Compared with the conventional classifiers, deep learning models produce more accurate results in the processing of hyperspectral images containing high spectral content [8]. Owing to the improvement in computer power, the utilization of deep learning techniques has recently emerged and vastly spread in the field of remote sensing.

The process of deep learning is similar to the working principle of artificial neural networks (ANNs). However, deep learning structures consist of a lot of multiple layers, so the system is more complex than standard neural networks (e.g., multilayer perceptrons). ANNs resemble the human brain so that it contains

*Correspondence: kavzoglu@gtu.edu.tr

This work is licensed under a Creative Commons Attribution 4.0 International License.

artificial nerve cells to learn the underlying relationships between input and output data [9]. Until today, neural networks have been employed for many problems in remote sensing, including classification, object detection, and feature extraction [10–13]. Shallow neural networks have been developed for scientific and commercial purposes; however, they have not adequately performed the learning process to solve complex problems [14]. The term 'deep learning' refers to the structure that includes multiple layers using nonlinear processing for applications [15]. Therefore, the algorithm has a deep capacity to solve complex problems [16].

Convolutional neural networks (CNNs), a special type of multilayer neural network, are the most popular deep learning architectures [17]. The most fundamental property of CNNs that distinguishes them from the other neural networks is the use of convolutional filters. Their capacity can be controlled by varying their depth, and they also make strong and mostly correct assumptions about the nature of images [5]. Lately, CNNs have been employed in various applications including image classification. For example, in a recent study [18], a task-special CNN model was proposed for scene classification. A gradient descent optimization algorithm was used to discover suitable parameters in neural networks using UC Merced Land-Use, AID, and NWPU-RESISC45 datasets. The performance of CNN models outperformed other manual-designed CNN models. In another study [19], a CNN model based on the principle of the pixel-based classification algorithm was suggested. The training was started with a set of incorrect data and then, a small amount of labelled data was used for the process of classification. Finally, the classification process was conducted. It was observed that the learning approach performed better classification accuracy compared to the other tested ones. On the other hand, the extraction of spectral, spatial, and spectral-and-spatial-based deep features were investigated using a CNN model [15]. Hyperspectral images were classified using deep learning methods. Also, regularization techniques of the L2 parameter and dropout were employed in the deep learning model to prevent overfitting.

It has been reported that the patch size of samples significantly affects the performance of the deep learning process [20, 21]. According to [22], choosing the appropriate input patch size for the deep learning model is crucial because small patches cannot contribute enough knowledge for the model to identify characteristics, whilst huge patches can lead to semantically confused knowledge, influencing the prediction of CNN model. In a recent work [23], a patch-based CNN model was suggested, and also the performance of the model was compared to those of conventional classification methods. In a study [24], a patch-based classification produced better results in the detection of specific weeds using the CNN method. Also, feature learning and extraction methods were evaluated to assess the accuracy of a weed classification. Results showed that the performance of CNN architecture was sensitive to the selection of patch size. The effects of patch sizes on the classification of land cover were also studied [25]. For the aim of the study, image objects were selected as disordered polygons, but the input of the deep learning model was an orderly patch. The approach for determining the optimal patch sizes has a major impact on the classification results.

The main purpose of this paper is to conduct a performance analysis for 2D and 3D CNN models using the hyperspectral images of ROSIS sensor and Jilin-1 GP01 by considering different numbers of patches and sample sizes. To be exact, 2D and 3D CNN models were created and then applied to produce thematic maps using varying patch sizes (i.e. 9×9, 15×15, 21×21, 27×27, and 33×33) and sample sizes (i.e. 50, 100, 200, 400, and 600). In this manner, the combinatorial effect of patch and sample sizes was examined for two hyperspectral datasets in this research study. The novelty of this paper underlies the investigation of combinatory utilization of the different patch and sample sizes for employing Jilin-1 GP01 and reflective optics system imaging spectrometer (ROSIS) hyperspectral dataset in image classification using the 2D and

3D CNN algorithms. Furthermore, the evaluation of thematic maps produced by deep learning models was conducted based on accuracy measures (overall accuracy, Kappa coefficient, and F-score). McNemar's test was also utilized to examine whether the performance differences of the CNN models used to produce thematic maps with different patch and sample sizes are statistically significant or not.

2. Test sites and datasets

In this study, to evaluate the effects of patches and sample sizes on the hyperspectral image classification of land use/land cover (LULC), two datasets, including a well-known Pavia University hyperspectral dataset (Dataset-1) recorded by the ROSIS sensor and Jilin-1 GP01 imagery (Dataset-2) were classified using deep learning algorithms. The datasets were selected to demonstrate the effectiveness of the proposed deep learning models. The size of the image in Dataset-1 is 610 × 340 with 1.3 m spatial resolution. The ROSIS sensor has a spectral range from 430 to 860 nm with 115 spectral bands, and the width of the bands is 4 nm. However, only 103 bands were utilized in the study due to the elimination of water absorption bands. The image scene contains nine major land cover classes of 42,776 pixels, namely asphalt, meadows, gravel, trees, painted metal sheets, bare soil, bitumen, self-blocking bricks, and shadows. On the other hand, Kırşehir Province, located in the Central Anatolian region of Turkey, was selected as the second study area. In preprocessing stage, atmospheric and geometric correction operations were applied to the 19-band Jilin-1 GP01 image. Subsequently, the Gram-Schmidt pan-sharpening algorithm was applied to resample 10- and 20-m bands to 5-m resolution. Considering the characteristics of the study area, five land use/cover (LULC) classes, namely impervious surface, soil bare, rock, forest and vegetation classes, were employed in the classification process to produce thematic maps (Figure 1).

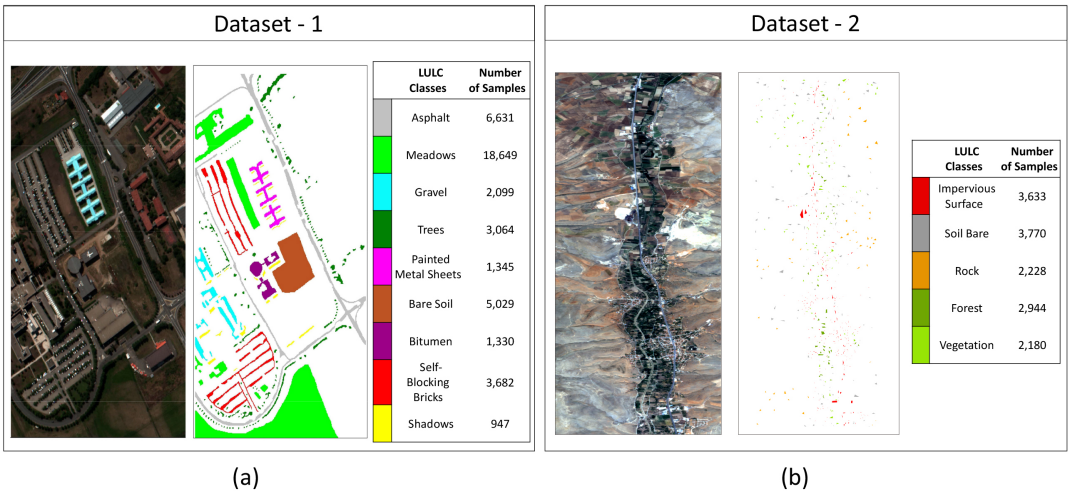

Figure 1. (a) Pavia University dataset (Dataset-1). (b) Kırşehir Dataset from Turkey (Dataset-2).

Before evaluating the effect of sample sizes on the performance of 2D and 3D CNN models, the validation datasets were constructed for both datasets from the labelled LULC with proportions of about 10% for evaluating the performance of deep learning models. In this manner, the training samples of 50, 100, 200, 400, and 600 were selected for each labelled class as balanced data to investigate the effect of the number of samples. Note that the process of selecting 50 samples was not applied to Dataset-2 because it caused overfitting problem. In the final stage, 250 samples per class were chosen as test dataset from the remaining dataset, which was used

to calculate the overall accuracies of all thematic maps. Besides, the effect of patch sizes using 9 × 9, 15 × 15, 21 × 21, 27 × 27, and 33 × 33 in pixel sizes on the deep learning model was examined in this research.

All processes in this study were performed in Python 2.7 using Anaconda's Jupyter Notebook. ERDAS Imagine and ArcGIS software packages were utilized to visualize and analyze the outputs produced in Python. The Keras library was preferred to create the CNN architecture for classifying a hyperspectral image. Finally, the Scikit-learn library was used for the accuracy analysis of the produced thematic maps. In this study, all processes for Dataset-1 were run on a personal computer with an Intel Core i9-10940X 3.30 GHz Central Processing Unit (CPU), 128 GB Random Access Memory (RAM), and NVIDIA GeForce RTX 2070 Graphics Processing Unit (GPU). On the other hand, a workstation (with an Intel Core i9-12900K 3.2 GHz CPU, 32 DDR4 RAM, and NVIDIA GeForce RTX 3090 GPU) was used for the analyses of Dataset-2.

3. Convolutional neural networks

In the literature, several deep learning models have been applied to remotely sensed imagery, including convolutional neural network (CNN), denoising autoencoder, deep belief network, long short-term memory, and recurrent neural network [26]. However, CNN has been the most commonly used structure for classification problems [6, 27, 28]. The most important difference that distinguishes this particular neural network model from the other networks is that each unit of layers in the network has one or more convolutional filters [29]. Various digital image processing tasks, namely preprocessing, image classification, segmentation, and object detection, can be successfully performed with CNN models. The capability of CNN stems from the capability of extracting the characteristics of large-scale data. For this reason, different architectures of CNN (e.g., LeNet, AlexNet, ResNet, SqueezeNet, MobileNEt and Xception) have been created for solving troublesome problems [30, 31].

The CNN model is a type of feed-forward neural network formed by various combinations of normalization, pooling, dropout, convolution, and flattening layers. It realizes the learning process by using a spatial local correlation by applying a connection model between neurons in these layers [30]. The convolutional layer is used to create a feature map. During the learning phase of the deep learning model, the weights of each filter are randomly selected. Convolutional filters perform operations by shifting the data. The pooling layer scales the output for the feature of large data by sampling and preserves the significant information. Thus, it reduces the processing load by using only important properties in the dataset. Internal covariate shift prevents convergence of deep learning model during training, so batch normalization is used [32]. Thus, it is used to improve the speed and performance of the model. The flattening layer converts a two-dimensional feature matrix into a one-dimensional vector. The use of all neurons in the layers of model can negatively affect the learning process. Therefore, the dropout layer is used to prevent this negative effect (i.e. avoid overfitting) [33].

A CNN model contains convolutional layers of different sizes, such as one, two, and three dimensional. A 1D convolutional layer is used to analyze basic structures, such as signal data [34]. It is also preferred for analyzing spectral or signal information. Thus, pixel-based classification can be performed using 1D CNN model in which each pixel corresponds to a spectral signature [16]. 2D convolutional filters can move in two directions and are most often used to process images or videos [35]. The advantage of the 2D convolutional layer is that it can extract both spectral and spatial information from the satellite imagery [36]. Therefore, learning models that include the 2D convolutional layer can produce solutions to the problems of object detection and classification. On the other hand, a 3D convolutional layer is applied to 3D datasets because the convolutional filter slides through 3D data. Moreover, 3D CNNs figure out features from multidimensional data, as in the case of spatio-

temporal images. Therefore, the 3D CNN is one of the essential architectures for learning demonstration for volumetric datasets [37, 38].

4. Results

4.1. Application of 2D CNN model

As a first step, the datasets created with different (50, 100, 200, 400, and 600) sample sizes and image patches (9 × 9, 15 × 15, 21 × 21, 27 × 27, and 33 × 33) were used for training the 2D CNN, except that the 50 samples could not be performed for Dataset-2 due to experienced overfitting problem. The deep learning model, consisting of 2D convolutional, flatting, dense, and dropout layers with an activation function, was designed for the current problems (Figure 2). The number of filters in the convolutional layers was chosen as eight after a trial-and-error approach. The unit parameters in the dense layer were set to 64 and 128. The ReLU activation function was used on all layers up to the output layer. In the final stage, Softmax activation was employed. During the training, the learning speed and decay parameters were selected as 0.001 and 1e-06, respectively. Categorical crossentropy was used in the cost function, and ADAM was also used in the optimization parameter [30]. The parameters of batch size were selected as 256 and 64.

No overfitting was observed in the models trained using the aforementioned training sample sets. For instance, the learning curves of 2D CNN models used in the analysis of different datasets with a combination of 400 samples and 15 × 15 patches are shown in Figure 3.

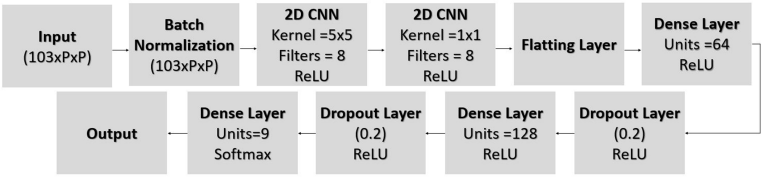

Figure 2. The structure of 2D CNN model applied for Dataset-1 and Dataset-2 in this study.

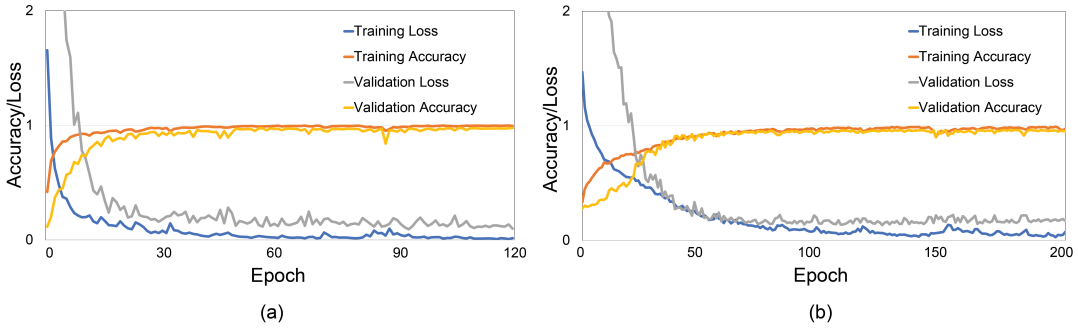

Figure 3. Learning curves of 2D CNN model using 15 × 15 patch size and 400 samples for (a) Dataset-1, and (b) Dataset-2.

After the training stage, thematic maps representing the LULC classes for Dataset-1 and Dataset-2 were produced with the trained 2D CNN model, respectively. The overall accuracies, Kappa statistics, and F-score values, derived from the harmonic means of the user and produces accuracies, were then calculated to evaluate overall and individual class accuracies (Tables 1 and 2).

Table 1. F-score values (%) of LULC classes, overall accuracies, and Kappa coefficients of Dataset-1 estimated for 2D CNN model using different patch sizes and number of samples.

Patch size	**9 × 9**					**15 × 15**					**21 × 21**				
LULC samples	**50**	**100**	**200**	**400**	**600**	**50**	**100**	**200**	**400**	**600**	**50**	**100**	**200**	**400**	**600**
Asphalt	84.74	91.55	92.84	96.84	95.88	81.83	86.34	94.23	95.90	98.59	82.30	86.90	90.97	93.84	95.88
Meadows	80.66	88.25	94.48	96.17	97.91	85.89	90.94	90.22	99.00	95.79	81.68	87.20	94.67	98.31	97.17
Gravel	83.86	85.99	94.19	95.02	97.32	74.70	81.87	91.94	96.37	97.94	75.62	87.04	93.30	95.52	97.16
Trees	93.96	98.81	98.90	99.11	99.30	98.76	96.42	99.25	99.58	99.10	97.67	98.32	99.00	99.42	98.29
Painted metal sheets	99.40	99.50	99.40	99.50	99.50	99.42	99.42	99.59	99.59	99.50	99.34	99.50	99.59	99.59	99.50
Bare soil	75.41	87.18	92.91	96.06	97.51	81.88	86.96	89.46	98.76	95.03	75.95	82.76	94.17	97.88	96.98
Bitumen	86.96	95.53	96.35	98.31	97.37	85.74	91.35	96.05	97.39	98.81	82.45	90.97	94.78	97.38	96.53
Self-blocking bricks	79.58	87.62	91.37	94.03	96.64	81.39	85.83	92.03	97.12	97.89	83.85	90.46	93.43	96.19	96.79
Shadows	98.20	98.91	98.81	99.01	99.11	98.85	98.44	99.09	99.26	99.11	96.84	97.17	99.01	98.77	99.11
Overall acc. (%)	87.40	93.07	95.96	97.60	98.33	88.06	91.22	95.06	98.52	98.47	86.63	91.57	95.85	97.85	97.98
Kappa coefficient	0.86	0.92	0.95	0.97	0.98	0.87	0.90	0.94	0.98	0.98	0.85	0.91	0.95	0.98	0.98
Time (s)	28.2	21.2	9.7	11.3	10.7	43.2	126.1	144.1	172.0	274.7	205.4	167.2	129.1	170.0	314.8

Table 1. (Continued).

Patch size	**27 × 27**					**33 × 33**				
LULC samples	**50**	**100**	**200**	**400**	**600**	**50**	**100**	**200**	**400**	**600**
Asphalt	85.18	90.77	93.94	96.49	97.48	85.98	89.92	95.34	97.04	99.20
Meadows	78.50	90.37	94.94	98.32	97.78	81.89	89.13	95.38	98.50	99.20
Gravel	85.26	91.22	96.78	97.47	98.72	91.14	94.76	98.52	99.20	99.50
Trees	97.99	97.47	98.49	99.42	98.90	94.39	96.51	98.29	99.30	99.20
Painted metal sheets	99.50	99.59	99.50	99.59	99.50	99.40	99.30	99.40	99.50	99.50
Bare soil	74.71	90.43	94.12	98.04	97.55	85.87	91.40	96.48	98.32	99.01
Bitumen	91.88	92.19	96.38	97.48	98.61	91.70	96.84	98.91	98.14	99.30
Self-blocking bricks	83.55	95.17	96.09	97.57	98.51	89.51	92.14	95.78	96.02	98.41
Shadows	97.35	97.72	99.09	99.26	99.11	95.08	96.89	98.62	99.11	99.01
Overall acc. (%)	88.61	94.28	97.00	98.59	98.96	91.09	94.62	97.91	98.84	99.64
Kappa coefficient	0.87	0.94	0.97	0.98	0.99	0.90	0.94	0.98	0.99	0.99
Time (s)	188.6	228.2	200.1	306.1	599.7	385.4	1248.7	1140.4	1865.3	1798.3

Table 2. F-score values (%) of LULC classes, overall accuracies, and Kappa coefficients of Dataset-2 estimated for 2D CNN model using different patch sizes and number of samples.

Patch size	**9 × 9**				**15 × 15**				**21 × 21**			
LULC samples	**100**	**200**	**400**	**600**	**100**	**200**	**400**	**600**	**100**	**200**	**400**	**600**
Impervious surface	91.02	88.10	94.23	92.08	89.93	93.06	97.68	97.78	86.42	91.25	95.59	96.00
Bare soil	92.10	93.96	97.12	98.77	92.47	96.14	96.5	99.15	92.11	94.00	97.40	98.57
Rock	90.23	91.22	96.14	97.90	91.02	94.07	97.15	97.60	89.60	93.26	97.35	98.26
Forest	92.08	91.42	97.39	95.35	89.22	96.29	94.46	96.82	91.48	94.96	97.87	98.01
Vegetation	84.51	82.30	89.94	92.47	81.45	92.00	91.30	97.20	86.25	90.01	94.00	96.85
Overall acc. (%)	89.21	89.7	94.33	95.86	89.02	94.42	95.42	96.87	88.90	92.39	96.01	97.30
Kappa coefficient	0.88	0.87	0.93	0.94	0.86	0.93	0.94	0.96	0.86	0.90	0.95	0.96
Time (s)	245.1	92.2	72.2	69.8	257.6	158.4	72.5	81.3	489.6	162.7	177.1	171.6

Table 2. (Continued).

Patch size	**27 × 27**				**33 × 33**			
LULC samples	**100**	**200**	**400**	**600**	**100**	**200**	**400**	**600**
Impervious surface	91.02	88.10	94.23	92.08	89.93	93.06	97.68	97.78
Bare soil	92.10	93.96	97.12	98.77	92.47	96.14	96.5	99.15
Rock	90.23	91.22	96.14	97.90	91.02	94.07	97.15	97.60
Forest	92.08	91.42	97.39	95.35	89.22	96.29	94.46	96.82
Vegetation	84.51	82.30	89.94	92.47	81.45	92.00	91.30	97.20
Overall acc. (%)	88.61	94.28	97.00	98.59	88.61	94.28	97.00	98.59
Kappa coefficient	0.87	0.94	0.97	0.98	0.87	0.94	0.97	0.98
Time (s)	188.6	228.2	200.1	306.1	385.4	1248.7	1140.4	1865.3

It can be seen from Table 1 that the overall accuracies were estimated as 87.40, 93.07, 95.96, 97.60, and 98.33% for the thematic maps produced with a 9 × 9 patch size using 50, 100, 200, 400 and 600 samples, respectively for Dataset-1. The highest F-score value of the LULC classes was calculated for the painted metal sheet class in thematic maps (over 99%). On the other hand, the lowest F-score value of LULC classes belongs to the classes of bare soil (75.41% for 50 samples), gravel (85.99% for 100 samples), and self-blocking bricks (91.37% for 200 samples and 94.03% for 400 samples). It should be also noted that the F-score values for the thematic map produced with 600 samples were over 95%. The accuracy assessment was conducted on the created thematic maps and overall accuracies of 88.06, 91.22, 95.06, 98.52 and 98.47% were produced for deep learning models with the 15 × 15 patch size using whole training samples, respectively. The highest F-score values were computed for the metal painted sheet class in the thematic maps produced with 200 and 400 samples (99.59%). On the other hand, the lowest F-score values (74.70 and 81.87%) were calculated for the gravel class in the thematic maps created with both 50 and 100 samples, respectively. It was also noticed that the gravel class was mainly mixed with asphalt, self-blocking bricks and bitumen classes. The lowest F-score value (89.46%) was estimated for the bare soil class using 200 samples, which was mainly mixed with the meadow class. It should be highlighted that the F-score values of all classes in the thematic maps produced with 400 and 600 samples were over 95%.

For the 21 × 21 patch size overall accuracies of the maps produced using 50, 100, 200, 400, and 600 samples were estimated as 86.63, 91.57, 95.85, 97.85, and 97.89%, respectively. The highest F-score accuracy was estimated for the painted metal sheet class (over 99%). The classes with the lowest F-score value were

computed for the class of gravel (75.62%) using 50 samples. Detailed analysis also revealed that the gravel class was mixed with the bare soil and self-blocking bricks classes. The lowest F-score value (82.76%) was estimated for the bare soil class using 100 samples per class. In the thematic maps produced with 200, 400, and 600 samples, the lowest F-score values (90.97, 93.84, and 95.88%) were calculated for the asphalt class, which was mixed with gravel, bitumen, and self-blocking bricks.

In the analysis of the 2D CNN model using 27 × 27 patch size, thematic maps produced with 50, 100, 200, 400, and 600 samples per class resulted in overall accuracies of 88.61, 94.28, 97.00, 98.59, and 99.11% and Kappa coefficients of 0.87, 0.94, 0.97, 0.98, and 0.99, respectively. The highest accuracies (over 99%) for LULC class were estimated for the painted metal sheet class. The LULC classes (i.e. meadow and bare soil classes) with the lowest F-score value were produced using 50 and 100 samples size in the thematic maps. It was noticed that these LULC classes were mostly mixed with each other. For 200, 400, and 600 samples, the lowest F-score value was produced for the asphalt class, which was mainly mixed with the bitumen class. The overall accuracies for the produced thematic maps with 33 × 33 patch size using 50, 100, 200, 400, and 600 training samples were calculated as 91.09, 94.62, 97.91, 98.84, and 99.64%, respectively. It is apparent that the highest F-score values of LULC classes were computed for the class of painted metal sheets (over 99.50%). Also, the lowest F-score values were calculated for the LULC class of meadows in the thematic maps produced with 50, 100, and 200 samples. It was also observed that the meadow class was mixed with the trees class. Also, the LULC class of self-blocking bricks had the lowest F-score values in the produced thematic maps with 400 and 600 samples.

Training samples including 100, 200, 400, and 600 for Dataset-2 was trained avoiding overfitting. The trained 2D CNN model was then used to generate thematic maps representing five LULC classes. The accuracy metrics were calculated to assess the overall and individual class accuracies (Table 2). The overall accuracies for the thematic maps created with a 9 × 9 patch size and 100, 200, 400, and 600 samples were estimated as 89.21, 89.70, 94.33, and 95.76%, respectively. When the individual LULC class accuracies were analysed using F-score values, it was observed that the bare soil class produced the highest F-score value for all patch sizes in the thematic maps whereas the lowest F-score value was estimated for vegetation class among all LULC classes (82.3%). For the 15 × 15 patch size employing 100, 200, 400, and 600 samples, the overall accuracies of 89.02, 94.42, 95.42, 98.52, and 96.87% were achieved, respectively. In the thematic maps produced with 600 samples and 15 × 15 patch size, the highest F-score value was estimated for the bare soil class (99.15%). Nonetheless, the lowest F-score values were computed for the vegetation class in the thematic maps generated with all samples. Detailed analysis of confusion matrices revealed that the vegetation class was mixed with the forest and bare soil classes. Moreover, the F-score values of all LULC classes were over 96% for 600 samples for the 21 × 21 patch size, the 2D CNN model with whole samples were calculated the overall accuracies as 88.90, 92.39, 96.01, and 97.30%, respectively. The highest F-score accuracies (over 94%) were estimated for the bare soil and forest classes. On the other hand, the lowest F-score values were computed for the vegetation class as 86.25, 90.01, and 94.00% using 100, 200, and 400 samples, respectively. The lowest F-score value was estimated for the impervious surface class using 600 samples with a 9 × 9 patch size.

In the evaluation of the deep learning model employing 27 × 27 patch size, thematic maps produced with 100, 200, 400, and 600 samples per LULC class yielded the overall accuracies of 87.71, 93.72, 96.35, and 96.38%, and Kappa coefficients of 0.85, 0.92, 0.95, and 0.95, respectively. The highest class-level accuracies were estimated for the rock class in the thematic maps produced for 100 (92.70%), 200 (97.01%), and 600 (98.20%) samples. Furthermore, the highest F-score value of the LULC class with 98% was calculated for the forest class

in the thematic maps using 400 samples. Besides, the vegetation class had the lowest individual class accuracy for 2D CNN model with 27 × 27 patch size. It was noticed that the vegetation class was again mixed with the bare soil and forest classes. In the analysis of 33 × 33 patch size, the overall accuracies of the thematic maps created with 100, 200, 400, and 600 samples were computed as 85.52, 87.53, 96.09, and 97.25%, respectively. It was seen that the highest F-score values (over 94%) were calculated for the forest class in the thematic maps whereas the lowest accuracy belongs to the impervious surface in all thematic maps, except for the thematic maps produced with 200 samples. In the thematic maps using 33 × 33 patch size, detailed spatial analysis for the impervious surface was performed with the lowest accuracy since the larger patch size reduced the spatial details in the thematic maps.

The thematic maps created with the highest accuracy for the dataset using 2D CNN models were given in Figure 4. For Dataset-1, the highest accuracy was calculated for 600 sample combinations with 33 × 33 patch size, whereas the highest accuracy was achieved with 21 × 21 patch size for 600 sample using Dataset-2. It was seen that the non-model parameters (i.e. sample and patch size) used in the model training varied according to type of dataset. In other words, thematic maps might not be produced with the highest accuracy with the largest patch and sample sizes.

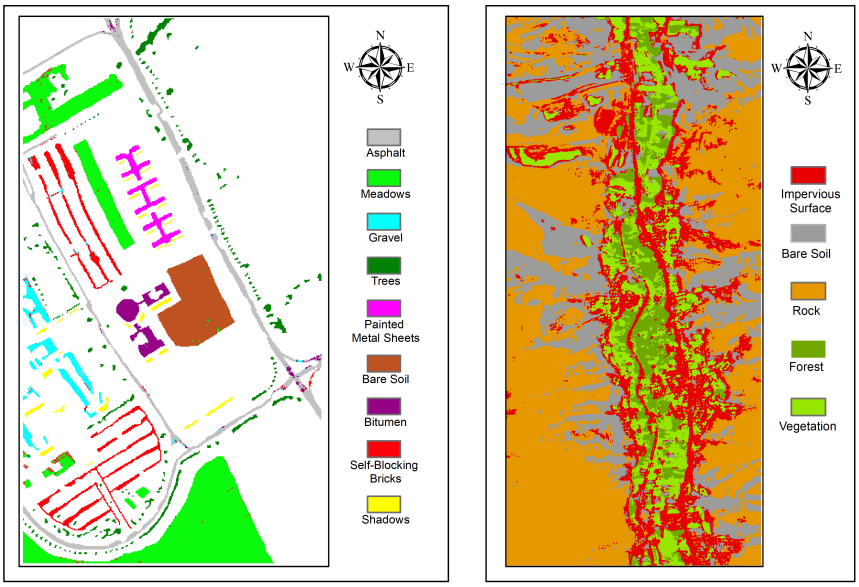

Figure 4. Thematic maps produced with the highest overall accuracy using different patch and samples sizes utilizing 2D CNN models. (a) Dataset-1. (b) Dataset-2.

Times required for training the 2D CNN models using Dataset-1 and Dataset-2 were also recorded and presented in Tables 1 and 2. It was observed that the training time got longer with the increased patch size, but irregular behaviour was observed when different sample sizes were applied for the same patch size. To be more specific, the fastest training time was recorded for 9 × 9 patch size with 600 samples and the slowest 33 × 33 patch size with 100 samples for both datasets. It could be stated that the change in training times depends on the interrelationship of patch sizes and the number of samples since these factors affect the computational complexity of the deep learning model in a combinatory way.

4.2. Application of 3D CNN model

Training datasets formed with different sample sizes were also tested with 3D CNN models using different patch sizes for Dataset-1 and Dataset-2. The proposed 3D CNN model includes several layers, including input, batch normalization, 3D convolutional, flatting, dense, and dropout layers (Figure 5). The ReLU function was used as an activation function in the last layer. The total number of parameter units in the dense layer was 521 for both cases. The parameters in the dropout layer were set to 20, 30, and 40%. The training process was completed by using the Softmax activation function. During the training phase, the learning speed was chosen as 0.001, and the decay parameter was 1e-06. Categorical cross-entropy was employed in the cost function, and ADAM optimization algorithm was chosen as an optimization parameter. The parameter of batch size was selected as 256.

No overfitting was observed during the training phase of 3D CNN models. Sample learning curves for both datasets were presented in Figure 6. The learning curve graph indicates model performance for the patch and sample size parameters, demonstrating all CNN models effectively associating links between the validation and training datasets. For whole training processes, validation and training loss values converged to about 0.1, while validation and training accuracy values approached 0.9.

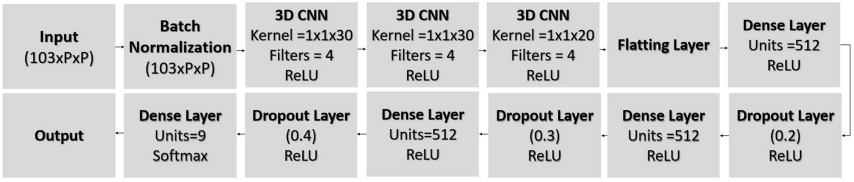

Figure 5. Structure of 3D CNN model adapted for Dataset-1 and Dataset-2 in this study.

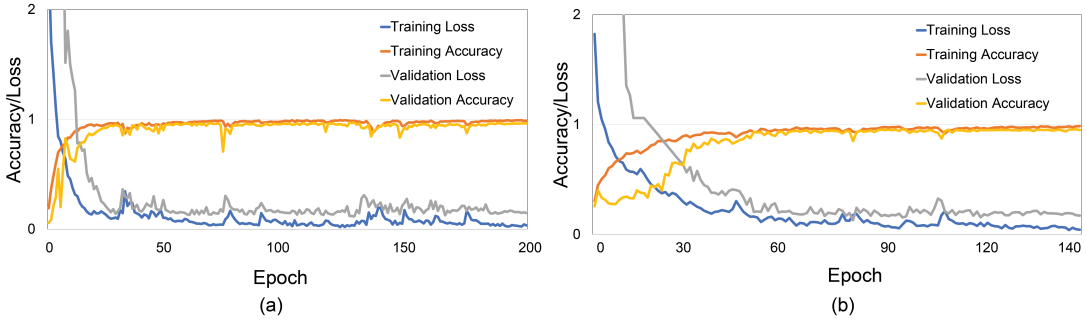

Figure 6. Learning curves of 3D CNN model using 15 × 15 patch size and 400 samples for (a) Dataset-1, and (b) Dataset-2.

Similar to the 2D CNN model, training datasets containing 9 × 9, 15 × 15, 21 × 21, 27 × 27 and 33 × 33 patches were constructed to analyze 50, 100, 200, 400, and 600 samples in the application of the 3D CNN model. Thus, 25 thematic maps representing nine LULC classes for Dataset-1 and 20 thematic maps representing five LULC classes for Dataset-2 were produced (Tables 3 and 4).

In the analysis of Dataset-1, the overall accuracies of 83.84, 94.58, 96.58, 97.73, and 98.67% were estimated for the 3D CNN model with 9 × 9 patch size using 50, 100, 200, 400, and 600 samples, respectively (Table 3). When the individual LULC class accuracies measured by F-score values were examined, it was observed that the highest F-score values (over 99%) were calculated for the LULC class of painted metal sheets. On the other hand, the lowest F-score values were estimated for the self-blocking bricks class (64.52% for 50 samples and

89.15% for 100 samples) and the asphalt class (89.48% for 200 samples). After applying the 3D CNN model using the 15 × 15 patch size for 50, 100, 200, 400, and 600 training samples, overall accuracies of 91.37, 95.72, 97.06, 99.35, and 99.44% were calculated, respectively.

Similar to the previous findings, the highest accuracy of LULC classes was estimated for the painted metal sheet class in the thematic maps. According to the F-score values, the lowest F-score values were estimated for the class of self-blocking bricks, which was mainly mixed with gravel, and asphalt classes, which was mixed with the bitumen and the self-blocking brick classes. The overall accuracies were calculated as 90.81, 96.24, 98.11, 99.52, and 99.51% using whole training samples for 21 × 21 patches, respectively. The painted metal sheet class had the highest individual class accuracy, and the lowest F-score value (79.93%) was calculated for the bare soil class. The classes of the self-blocking bricks and asphalt had the lowest F-score value ($\sim$ 91%) for 100 samples. While the asphalt class was mixed with the bitumen class, the brick class was mixed with the gravel class. The overall accuracies of the maps produced using different training samples (50, 100, 200, 400, and 600) were 87.17, 95.98, 97.33, 99.43, and 99.42% using a 27 × 27 patch, respectively. Shadow and painted metal sheets had the highest F-score accuracies (over 98%), but the gravel class had the lowest accuracy using 50 samples per class. For 400 and 600 training samples, the highest accuracy (over 99%) was estimated for gravel and painted metal sheet classes (Table 3). In the final part, the analysis of the 3D CNN model using 33 × 33 patch size was conducted and overall accuracies of the thematic maps created with 50, 100, 200, 400, and 600 training samples were calculated as 91.76, 95.49, 98.18, 99.40, and 99.51%, respectively. It was found that the highest F-score values were estimated for the painted metal sheets class for the produced thematic maps using 50, 100, and 200 samples (above 97%). On the other hand, the individual F-score values of all LULC classes of thematic maps produced with 400 and 600 were equal to or above 97.5%.

For Dataset-2 the overall accuracies of 91.15, 93.31, 95.12, and 95.31% were calculated for the 3D CNN model using 9 × 9 patch size utilizing all sample combinations, respectively (Table 4). The highest F-score values were computed for the rock class (i.e. 94.54% and 97.00% for 100 and 600 samples, respectively). Nevertheless, the highest accuracies were calculated for the bare soil class (98.35%) using 400 samples, and the forest class (95.46%) using 200 samples. It was also found that the vegetation class estimated the lowest F-score values (between 84 and 93%).

For the 3D CNN model with a 15 × 15 patch size utilizing 100, 200, 400, and 600 samples in Dataset-2, the overall accuracies of 91.40, 93.12, 94.98, and 96.82% were computed, respectively. The highest F-score value of LULC classes belongs to the class of bare soil (98.07% for 400 samples and 97.23 for 200 samples), impervious surface (93.59% for 100 samples), and vegetation (96.90% for 200 samples). Besides, the lowest F-score values (85.56, 86.00 and 94.00%) were estimated for the LULC vegetation class in the thematic maps created with both 100, 200 and 600 samples, respectively. Also, overall accuracies were computed as 91.32, 95.19, 96.60, and 96.93% for thematic maps created with a patch size of 21 × 21 utilizing 100, 200, 400, and 600 samples, respectively. According to the F-score values, the highest F-score values belong to the class of soil bare for the thematic maps using 100, 200 and 400 training samples and the class of rock for the thematic maps using 600 training samples. For all thematic maps produced with 21 × 21 patch size, the vegetation class yielded the lowest class-level accuracies.

Table 3. F-score values (%) of LULC classes, overall accuracies, and Kappa coefficients of Dataset-1 estimated for 3D CNN model using different patch sizes and number of samples.

Patch size	**9 × 9**					**15 × 15**					**21 × 21**				
LULC samples	**50**	**100**	**200**	**400**	**600**	**50**	**100**	**200**	**400**	**600**	**50**	**100**	**200**	**400**	**600**
Asphalt	84.78	90.60	89.48	97.31	99.10	86.67	91.11	94.13	97.89	98.28	87.38	90.77	95.03	99.50	99.50
Meadows	74.26	93.58	97.92	95.15	96.74	91.10	98.68	96.42	99.67	99.80	82.49	98.52	98.77	98.93	99.01
Gravel	73.25	90.82	95.79	97.80	98.80	83.28	89.70	95.28	98.83	98.50	89.68	91.87	97.65	99.09	98.91
Trees	95.54	98.09	97.59	99.01	97.56	97.11	98.83	98.75	99.67	99.40	96.71	98.41	98.07	98.74	98.80
Painted metal sheets	99.40	99.10	99.50	99.50	99.50	99.59	99.59	99.59	99.59	99.50	99.26	99.42	99.59	99.59	99.50
Bare soil	71.52	92.43	97.70	93.50	95.69	88.37	98.01	96.00	99.34	99.40	79.93	98.34	99.25	99.42	99.21
Bitumen	88.89	94.04	92.78	97.61	99.30	92.27	93.74	96.39	98.93	99.11	93.40	95.53	96.37	99.26	99.30
Self-blocking bricks	64.52	89.15	94.80	96.33	97.93	82.76	89.20	94.20	98.35	98.13	86.96	90.94	95.47	98.16	97.80
Shadows	99.01	99.11	99.11	98.91	99.01	97.79	99.01	99.17	98.20	98.42	97.94	98.51	99.09	99.26	99.11
Overall acc. (%)	83.84	94.58	96.58	97.73	98.67	91.37	95.72	97.06	99.35	99.44	90.81	96.24	98.11	99.52	99.51
Kappa coefficient	0.82	0.94	0.96	0.97	0.98	0.90	0.95	0.97	0.99	0.99	0.90	0.96	0.98	0.99	0.99
Time (s)	295.8	668.3	282.5	595.3	539.5	362.7	302.3	101.1	198.0	642.8	484.4	321.0	202.1	226.1	522.2

Table 3. (Continued).

Patch size	**27 × 27**					**33 × 33**				
LULC samples	**50**	**100**	**200**	**400**	**600**	**50**	**100**	**200**	**400**	**600**
Asphalt	75.60	92.23	90.58	98.33	98.21	87.87	89.94	94.87	99.00	99.00
Meadows	85.52	92.50	99.17	98.50	98.49	83.30	94.74	98.81	99.50	99.70
Gravel	70.07	98.04	96.67	99.50	99.01	92.85	94.33	97.18	99.50	98.72
Trees	94.23	97.98	97.84	98.91	99.40	95.23	96.93	98.70	99.30	99.40
Painted metal sheets	99.09	99.59	99.59	99.59	99.50	97.94	99.30	99.50	98.80	99.50
Bare soil	84.89	92.02	99.34	99.42	99.40	86.60	92.62	97.45	99.40	98.90
Bitumen	91.12	96.00	95.16	98.77	98.72	88.76	96.91	99.21	98.91	99.30
Self-blocking bricks	81.01	94.33	95.14	98.84	98.50	90.76	93.40	94.38	98.20	97.51
Shadows	98.52	97.44	98.44	99.26	99.11	97.71	96.82	99.01	97.56	99.11
Overall acc. (%)	87.17	95.98	97.33	99.43	99.42	91.76	95.49	98.18	99.40	99.51
Kappa coefficient	0.86	0.95	0.97	0.99	0.99	0.91	0.95	0.98	0.99	0.99
Time (s)	1001.5	657.7	234.2	345.1	946.0	6400.4	11567.1	6225.4	25160.8	17550.3

Table 4. F-score values (%) of LULC classes, overall accuracies, and Kappa coefficients of Dataset-2 estimated for 3D CNN model using different patch sizes and number of samples.

Patch size	**9 × 9**				**15 × 15**				**21 × 21**			
LULC samples	**100**	**200**	**400**	**600**	**100**	**200**	**400**	**600**	**100**	**200**	**400**	**600**
Impervious surface	91.10	92.56	93.80	94.28	93.59	93.84	93.15	96.15	90.02	95.47	96.55	97.00
Bare soil	94.26	95.30	98.35	96.21	93.00	97.23	95.40	98.07	96.15	98.00	98.10	97.98
Rock	94.55	92.02	96.67	97.00	92.36	96.52	94.02	97.03	95.00	97.25	97.47	99.15
Forest	91.52	95.46	95.21	95.80	91.03	92.05	94.78	96.90	91.75	95.21	98.05	97.30
Vegetation	84.13	91.37	92.16	93.40	85.56	86.00	96.90	94.00	84.09	89.96	95.61	93.20
Overall acc. (%)	91.15	93.31	95.12	95.31	91.40	93.12	94.98	96.82	91.32	95.19	96.60	96.93
Kappa coefficient	0.89	0.92	0.93	0.94	0.89	0.91	0.92	0.95	0.89	0.93	0.95	0.96
Time (s)	1007.7	1046.6	1043.8	1064.0	1022.2	1071.6	1022.6	1028.9	1023.9	1076.5	1082.8	1097.6

Table 4. (Continued).

Patch size	**27 × 27**				**33 × 33**			
LULC samples	**100**	**200**	**400**	**600**	**100**	**200**	**400**	**600**
Impervious surface	89.42	90.35	97.14	95.23	83.25	85.17	94.66	97.00
Bare soil	94.42	95.79	97.90	96.90	90.70	95.50	98.34	98.90
Rock	96.20	99.00	98.55	98.03	95.45	97.07	98.10	99.95
Forest	93.02	97.03	97.25	98.86	94.00	95.30	97.44	99.30
Vegetation	82.41	88.38	94.00	95.44	85.68	83.11	95.72	99.25
Overall acc. (%)	90.90	93.67	96.90	96.80	89.18	90.89	96.87	98.33
Kappa coefficient	0.89	0.92	0.95	0.96	0.86	0.89	0.96	0.97
Time (s)	1002.9	1051.7	1101.2	1171.0	714.1	904.9	1157.2	1173.9

To evaluate performance of the applied DL models using 27 × 27 patch size, the overall accuracies of the thematic maps generated with 100, 200, 400, and 600 sample sets were calculated as 90.90, 93.67, 96.90, and 96.80%, respectively. It was concluded that the highest F-score values were assessed for the rock class for the thematic maps utilizing 100, 200, and 400 samples (above 96%) and the vegetation class for the thematic maps created with 27 × 27 patch size using 600 samples (98.86%). In the classification processes of 33 × 33 patch settings, the overall accuracies were estimated as 89.18% for 100 samples, 90.89% for 200 samples, 96.87% for 400 samples, and 98.33% for 600 samples. It was observed that the highest F-score values were estimated for the rock class for the generated thematic maps using 100 and 200 samples (above 95%), whereas the lowest F-score accuracies were computed for the impervious surface and vegetation classes in the thematic maps using 100 and 200 samples, respectively.

The thematic maps with the highest overall accuracy, produced with 3D CNN models using varying patch and sample sizes, are shown in Figure 7. The highest accuracy was computed for 400 sample variations with a 21 × 21 patch size for Dataset-1, and the highest accuracy was calculated for 600 sample combinations with a 33 × 33 patch size for Dataset-2.

The training times for the 3D CNN model acquired for Dataset-1 and Dataset-2 are also shown in Table 3 and 4. For the Pavia hyperspectral dataset (Dataset-1), the execution time for 33 × 33 patch size with 400 samples had the highest run time (25,160.8 s). Also, the training for 15 × 15 patch size with 200 samples ended up with the shortest run time (101.1 s). In the classification of Jilin-1 GP01 imagery (Dataset-2), the training time for 33 × 33 patch size with 600 training samples resulted in the longest run time (1173.8 s), whereas the period of training for 33 × 33 patch size with 100 samples took significantly shorter time (714.136

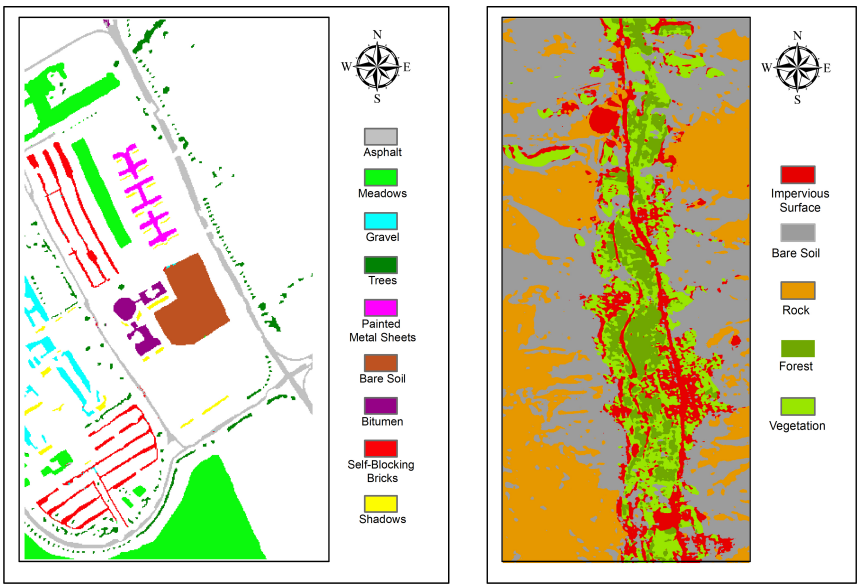

Figure 7. Thematic maps produced with the highest overall accuracies using different patch sizes and number of samples utilizing 3D CNN model for (a) Dataset-1, and (b) Dataset-2.

s). Results are parallel to those of Dataset-1 where the average training duration for the patches increased with the number of samples per class. Moreover, these results revealed that patch size and the number of samples had a considerable effect on training durations, but their effect on training times was nonlinear.

5. Discussion

It is critical to determine optimal values for model (i.e. the number of layers, filter size, epoch number and learning rate) and non-model (i.e. sample and patch size) parameters in the deep learning classification process. Despite the limited examination of the effect of patch or sample sizes on the classification performance, to the best of the authors' knowledge, the effect of both the training sample and patch size on the classification accuracy has not been studied extensively in the literature. Therefore, the effect of combining different samples and patch sizes on the deep learning models (i.e. 2D and 3D CNN) was investigated on two different hyperspectral images in this study. To deduce conclusions from the results, overall accuracies estimated for the thematic maps produced with 2D and 3D CNN models were summarized in Figure 8. A significant effect of image particles and sample sizes on the training of the deep learning model was observed. It was clear that the thematic map accuracy improves with the increase in the number of samples per class until a certain point (400 samples here), achieving the best overall accuracies for 400 and 600 samples per class for both datasets. Furthermore, the accuracy of the thematic maps produced with all patches showed variation depending on the selected number of training samples. More specifically, the 2D CNN model with 600 samples culminated in a remarkable increment in overall accuracies ($\sim 10\%$) compared to 50 samples per class in the case of Dataset-1. On the other hand, the estimated accuracies were higher for the thematic maps with 600 samples compared to the thematic maps with 100 samples using 2D CNN for Dataset-2. For the 3D CNN model applied to Dataset-1, the highest overall accuracies were estimated for all patches except for the 9x9 patch size using 400 and 600 samples (over 99%).

Moreover, overall accuracy of over 95% was obtained for the thematic maps produced with Dataset-2 used for 3D CNN model with 600 samples.

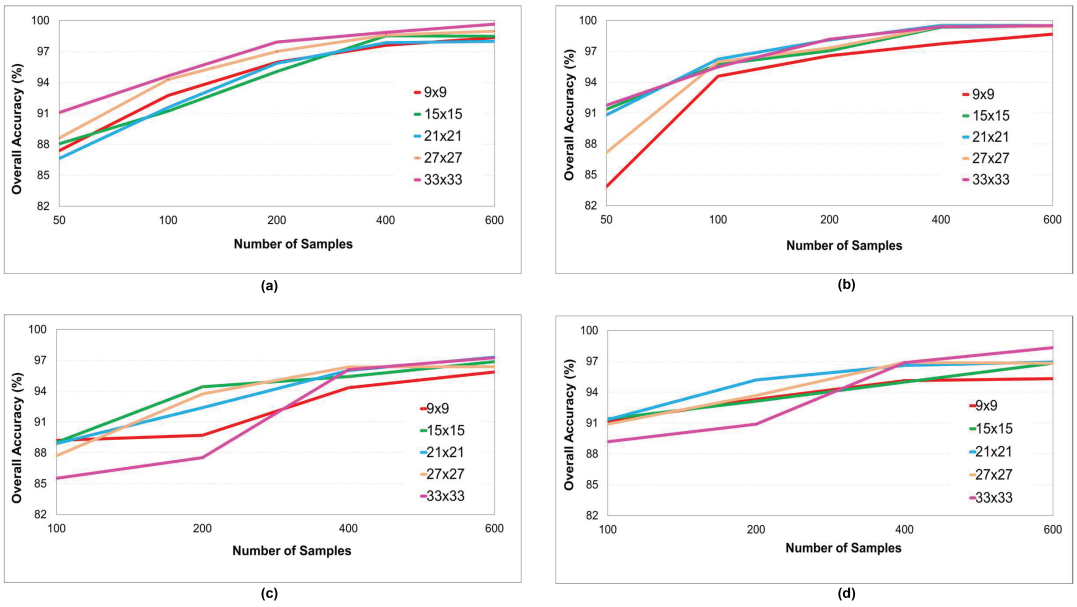

Figure 8. Overall accuracies calculated for different number of samples considering different patch sizes using a) 2D CNN model for Dataset-1, b) 3D CNN model for Dataset-1, c) 2D CNN model for Dataset-2, and d) 3D CNN model for Dataset-2.

Another important finding is that patch size becomes an important factor when limited training samples are employed in model formation. In the examination of Dataset-1 using the deep learning models, as in the case of 50 samples per class, whilst the use of 9 × 9 patch produced inferior results (an overall accuracy of 87.40% for the 2D CNN and 83.84% for the 3D CNN), using 33 × 33 patch size resulted in high accurate results (overall accuracies of 91.09% for the 2D CNN and 91.76% for the 3D CNN models). On the other hand, it was determined that 2D and 3D CNN models were overfitted on Dataset-2 using 50 samples. Thus, only 100, 200, 400 and 600 samples were used for the analysis of classification. Employing 100 samples in the investigation of Dataset-2, 9 × 9 patch provided an overall accuracy of 89.2% for the 2D CNN model and 91.15% for the 3D CNN model whilst employing a 33 × 33 patch size produced overall accuracies of 85.52% for the 2D CNN and 89.18% for the 3D CNN models. For larger training samples (400 and 600 samples) similar performances were observed for the patches of Dataset-1 and Dataset-2.

The results of this study also revealed that the change of patch size led to striking effects on the performance of the deep learning models. This finding supports previous research [21] that increasing patch size allows the involvement of spectral information together with more spatial information obtained from adjacent pixels. Therefore, it can extract more information during the training deep learning models. In a recent study, [39] revealed that higher accuracy was achieved with huge patch size in the classification of the Pavia dataset using deep learning algorithms. However, the influence of patch size selection on the classification accuracy differs according to the model with different characteristics and the type of remotely sensed image [40]. In conclusion, it can be concluded that the effect of non-model parameters on the classification accuracy mainly depends on both the dataset and the architecture of the model. In other words, the change in patch and sample

sizes in the classification of datasets with different spectral characteristics affects the thematic map accuracy. In addition, findings were obtained to support the outcomes of aforementioned studies. For instance, increasing the sample size with a fixed patch size causes an increase in computational cost. The classified satellite image with the high accuracy can be produced with a large patch size together with a small number of samples.

To compare the model performances, McNemar's statistical test, the most widely used method in remote sensing studies, was employed in this study [41, 42]. The test, which is based on a crosstabulation of the classification results, was applied to the results to be able to state that the performance differences are statistically significant or not. If the estimated value is greater than the critical table value, there is a significant difference between the results produced by different models. The test was performed here to determine how the number of samples and patch sizes selected for training affect the performances of the proposed 2D and 3D CNN models. The test was performed at a 95% confidence interval, so it was examined whether the values obtained by testing thematic maps were greater than the critical table value of 3.84 in the χ^2 distribution table. Test results revealed that the classification outcomes of Dataset-1 which were produced using 2D and 3D CNNs excluding 400 and 600 samples for all patches were statistically significant since the estimated values were greater than the critical value (Table 5). Thus, it was proved that the number of samples caused a significant increase in the thematic map accuracies, particularly for a lower number of samples. McNemar's test was also applied to the results produced for Dataset-2 using different patch and sample sizes. Results were similar to those of Dataset-1, except for both the 15 × 15 patch size and 400 samples used in the 2D CNN model, and the 33 × 33 patch size and 200 samples used in the 3D CNN model (Table 5).

For all patch sizes for Dataset-1, the highest test values were calculated between 50 and 400/600 samples in the applications of the 2D and 3D CNN models. This result validates the positive effect of the number of training samples on the accuracy of deep learning-based classification. Furthermore, the lowest McNemar's test values were estimated between 400 and 600 samples, indicating similar performances; thus results indicate the sufficiency of number of samples for learning.

To analyze the effect of patch size selection, McNemar's test was also performed on the produced results (Table 6). In the case of 50 samples per class, significant improvement in classification performance was only achieved with the use of 33x33 patches for the 2D CNN model. On the other hand, 9 × 9 patch size was ineffective for learning the problem, except for 600 samples per class, since the performance differences with the other patch sizes were found statistically significant for the 3D CNN models.

In the utilization of 2D and 3D CNN models, the highest McNemar's values were estimated for 200 training samples among all patch sizes in Dataset-2 (Table 6). In addition, the lowest McNemar's test values were calculated for the thematic maps using 600 training samples in 2D CNN model and 200 training samples in 3D CNN model. In 2D CNN model applications, it can be stated that the 33 × 33 patch size had a significant effect on the overall accuracy for 100 and 200 samples. When the changes in the application results including 600 samples used in 2D CNN model according to the patch size were examined in detail, it could be said that the change of the patch size did not affect the results significantly, except for the McNemar's test value with 6.37. According to Table 6, it was confirmed by the McNemar's test that increasing the patch size had a significant positive effect on the results produced by using less training samples in the 2D CNN model. Larger patch sizes (i.e. 33 × 33) contributed to statistically significant improvements in the classification performance of the 2D CNN model. Also, similar performances were produced when 400 and 600 samples were employed in the training stage.

Table 5. McNemar's test results for the thematic maps produced with 2D and 3D CNN models using different patch sizes. Note that computed statistics greater than the critical table value ($\chi^2 = 3.84$) presented in bold value.

		2D CNN							3D CNN						
Patch size	No. of samples	Dataset-1				Dataset-2			Dataset-1				Dataset-2		
		100	200	400	600	200	400	600	100	200	400	600	200	400	600
9 × 9	50	**12.19**	**17.59**	**21.96**	**22.44**	**-**	**-**	**-**	**19.11**	**22.59**	**24.66**	**26.19**	**-**	**-**	**-**
	100	-	**6.36**	**12.17**	**13.76**	**3.59**	**6.83**	**5.70**	-	**6.16**	**8.67**	**13.21**	**7.69**	**8.40**	**5.85**
	200	-	-	**5.56**	**8.60**	**-**	**4.56**	**5.66**	-	-	**3.91**	**8.37**	-	**3.88**	**3.85**
	400	-	-	-	3.51	**-**	**-**	3.61	-	-	-	3.58	-	-	2.09
15 × 15	50	**6.19**	**14.43**	**22.92**	**20.95**	**-**	**-**	**-**	**9.83**	**14.41**	**19.64**	**21.27**	**-**	**-**	**-**
	100	-	**7.48**	**18.63**	**16.49**	**9.33**	**12.03**	**12.72**	**-**	**5.34**	**12.69**	**12.57**	**7.10**	**8.09**	**5.24**
	200	-	-	**10.90**	**8.21**	**-**	1.44	**3.54**	**-**	**-**	**8.36**	**8.38**	**-**	**4.44**	**8.32**
	400	-	-	-	3.21	**-**	**-**	0.65	**-**	**-**	**-**	0.71	**-**	**-**	7.91
21 × 21	50	**8.92**	**18.41**	**22.41**	**21.46**	**-**	**-**	**-**	**15.71**	**16.79**	**22.08**	**21.20**	**-**	**-**	**-**
	100	**-**	**10.84**	**15.79**	**15.36**	**5.75**	**10.83**	**10.36**	**-**	**4.45**	**11.68**	**11.64**	**7.73**	**4.00**	**3.89**
	200	**-**	**-**	**7.75**	**7.23**	**-**	**4.39**	**7.47**	**-**	**-**	**8.07**	**8.28**	**-**	**4.16**	**3.87**
	400	**-**	**-**	-	0.50	**-**	**-**	3.04	**-**	**-**	**-**	0.71	**-**	**-**	2.39
27 × 27	50	**11.77**	**18.95**	**22.87**	**22.58**	**-**	**-**	**-**	**18.33**	**20.28**	**25.11**	**22.81**	**-**	**-**	**-**
	100	**-**	**7.76**	**13.24**	**14.79**	**8.84**	**9.38**	**7.97**	**-**	**5.47**	**12.20**	**11.48**	**5.26**	**8.26**	**7.89**
	200	**-**	**-**	**6.15**	**7.11**	**-**	**4.24**	**5.20**	**-**	**-**	**8.70**	**8.46**	**-**	**4.87**	**9.49**
	400	**-**	**-**	**-**	1.64	**-**	**-**	2.93	**-**	**-**	**-**	0.50	**-**	**-**	3.27
33 × 33	50	**8.05**	**16.83**	**19.13**	**20.52**	**-**	**-**	**-**	**9.28**	**15.85**	**19.55**	**19.20**	**-**	**-**	**-**
	100	**-**	**10.90**	**14.53**	**15.41**	**3.99**	**8.19**	**6.61**	**-**	**4.64**	**9.66**	**11.99**	1.77	**7.93**	**13.42**
	200	**-**	**-**	**4.46**	**7.67**	**-**	**7.19**	**5.67**	**-**	**-**	**6.77**	**5.59**	**-**	**7.09**	**9.85**
	400	**-**	**-**	**-**	3.58	-	-	2.54	-	-	-	0.77	-	-	2.80

Table 6. McNemar's test results for the thematic maps produced with 2D and 3D CNN models using different numbers of training samples in the case of Dataset-1 and Dataset-2. Note that computed statistics greater than the critical table value ($\chi^2 = 3.84$) presented in bold value.

		2D CNN								3D CNN							
No. of samples	Patch size	Dataset-1				Dataset-2				Dataset-1				Dataset-2			
		15 × 15	21 × 21	27 × 27	33 × 33	15 × 15	21 × 21	27 × 27	33 × 33	15 × 15	21 × 21	27 × 27	33 × 33	15 × 15	21 × 21	27 × 27	33 × 33
50	9 × 9	0.10	0.89	1.20	**6.56**	-	-	-	-	**11.99**	**12.20**	**6.38**	**12.24**	-	-	-	-
	15 × 15	-	1.90	0.51	**6.37**	-	-	-	-	-	1.44	**6.67**	**4.17**	-	-	-	-
	21 × 21	-	-	2.38	**7.87**	-	-	-	-	-	-	**5.80**	**4.00**	-	-	-	-
	27 × 27	-	-	-	**5.97**	-	-	-	-	-	-	-	**7.98**	-	-	-	-
100	9 × 9	**4.80**	**4.75**	**4.45**	**7.58**	1.05	2.40	**13.52**	**16.12**	**4.12**	**6.47**	**4.16**	**3.87**	2.54	1.51	3.54	**9.12**
	15 × 15	-	0.90	**5.59**	**7.55**	-	**5.97**	**4.03**	**7.99**	-	3.75	1.30	1.98	-	2.65	**6.24**	**14.57**
	21 × 21	-	-	**5.38**	**6.15**	-	-	**11.90**	**13.11**	-	-	1.27	2.36	-	-	2.28	**12.37**
	27 × 27	-	-	-	1.03	-	-	-	**5.11**	-	-	-	2.06	-	-	-	**10.97**
200	9 × 9	3.31	1.37	3.41	**5.87**	**14.54**	**6.62**	**4.72**	**11.90**	2.28	5.57	2.94	**4.97**	1.54	**5.18**	1.39	**16.94**
	15 × 15	-	2.11	**5.06**	**8.89**	-	**7.93**	**4.36**	**12.36**	-	3.77	1.46	3.81	-	**4.76**	0.69	**10.50**
	21 × 21	-	-	3.21	**8.05**	-	-	1.74	**8.23**	-	-	1.37	1.32	-	-	**9.08**	**14.77**
	27 × 27	-	-	-	3.40	-	-	-	**8.92**	-	-	-	2.21	-	-	-	**7.88**
400	9 × 9	1.51	0.79	0.69	2.79	2.42	**5.30**	**5.31**	**7.18**	**4.68**	**8.74**	**6.56**	**6.12**	0.71	**4.38**	**7.90**	**9.81**
	15 × 15	-	1.33	0.17	1.63	-	2.60	**3.90**	3.27	-	1.30	0.63	0.71	-	3.08	**3.95**	3.18
	21 × 21	-	-	3.22	**4.75**	-	-	2.24	3.68	-	-	0.88	1.46	-	-	3.57	3.73
	27 × 27	-	-	-	2.48	-	-	-	3.59	-	-	-	1.53	-	-	-	2.05
600	9 × 9	2.44	2.49	0.78	3.15	3.82	**6.37**	2.45	3.27	3.12	3.69	3.66	2.44	2.35	1.66	2.26	**6.11**
	15 × 15	-	2.24	2.58	3.62	-	2.91	0.88	1.69	-	3.18	0.38	1.31	-	2.34	1.26	**7.90**
	21 × 21	-	-	**4.03**	**5.14**	-	-	0.39	1.94	-	-	2.00	0.57	-	-	1.36	**6.47**
	27 × 27	-	-	-	2.97	-	-	-	3.52	-	-	-	0.13	-	-	-	**4.87**

6. Conclusion

In this study, 2D and 3D CNN models were applied with different sample and patch sizes for the classification of hyperspectral datasets. For the classification of the Pavia (Dataset-1) and Jilin-1 GP01 (Dataset-2) hyperspectral images, training datasets were created by using 9 × 9, 15 × 15, 21 × 21, 27 × 27, and 33 × 33 patch sizes. Then, they were separately applied to 2D and 3D CNN models using 50 (except for the Jilin-1 GP01 image), 100, 200, 400, and 600 samples per class to find out the combined effect of both parameters on the produced classification accuracies. Although the training times of the 3D CNN models were significantly higher, they often produced higher accuracies than the 2D CNN models under the same setting. It was also observed that the changes in the size of these patches slightly affected the accuracy of the thematic map in comparison to the effect of training samples. In most cases, 9 × 9 patches for the datasets performed significantly poorer performances compared with the other patch choices. Similar results were produced for 50 training samples per class for Dataset-1 and for 100 training samples per class for Dataset-2. On the other hand, the size of the particles belonging to the training data and the architecture of the deep learning model have a direct effect on the training time. However, the sample size has a variable effect on classification accuracy regardless of the selected patch size. Therefore, it can be concluded that the change of image particle size depends on the deep learning model and selection of the dataset. In the scarcity of training samples, the use of a larger patch size can be suggested according to the results produced in this study. McNemar's test results revealed that the increasing number of training samples plays a significant role in the classification performances since both the 2D and 3D CNN models performed statistically more robust performances when the sample size was increased. Having less effect on CNN model performances, the selection of patch size was found to be critical for the datasets with limited training samples. Therefore, statistical test values were significantly different for the results produced with the largest patch sizes in this study. In other words, CNN models could produce more accurate results when large patch sizes are employed for limited training samples. Otherwise, statistically similar results in terms of classification accuracy can be obtained.

References

[1] Zhang L, Zhang L, Du B. Deep learning for remote sensing data: A technical tutorial on the state of the art. IEEE Geoscience and Remote Sensing Magazine 2016; 4 (2): 22–40. doi: 10.1109/MGRS.2016.2540798

[2] Kavzoglu T, Reis S. Performance analysis of Maximum Likelihood and Artificial Neural Network classifiers for training sets with mixed pixels. GIScience and Remote Sensing 2008; 45 (3): 330–342. doi: 10.2747/1548-1603.45.3.330

[3] Xie Z, Chen Y, Lu D, Li G, Chen E. Classification of land cover, forest, and tree species classes with ZiYuan-3 multispectral and stereo data. Remote Sensing 2019; 11 (2): 164. doi: 10.3390/rs11020164

[4] Shao Y, Lunetta RS. Comparison of Support Vector Machine, Neural Network, and CART algorithms for the land-cover classification using limited training data points. ISPRS Journal of Photogrammetry and Remote Sensing 2012; 70: 78–87. doi: 10.1016/j.isprsjprs.2012.04.001

[5] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with Deep Convolutional Neural Networks. Communications of the ACM 2017; 60 (6): 84–90. doi: 10.1145/3065386

[6] Mou L, Ghamisi P, Zhu XX. Deep Recurrent Neural Networks for hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing 2017; 55 (7): 3639–3655. doi: 10.1109/TGRS.2016.2636241

[7] Li S, Song W, Fang L, Chen Y, Ghamisi P et al. Deep learning for hyperspectral image classification: An overview. IEEE Transactions on Geoscience and Remote Sensing 2019; 57 (9): 6690–6709. doi:10.1109/TGRS.2019.2907932

[8] Wang ZY, Xia QM, Yan JW, Xuan SQ, Su JH et al. Hyperspectral image classification based on spectral and spatial information using multi-scale ResNet. Applied Sciences 2019; 9 (22): 4890. doi: 10.3390/app9224890

[9] Kavzoglu T. Increasing the accuracy of neural network classification using refined training data. Environmental Modelling & Software 2009; 24 (7): 850-858. doi:10.1016/j.envsoft.2008.11.012

[10] Nogueira K, Penatti OA, Santos dJA. Towards better exploiting Convolutional Neural Networks for remote sensing scene classification. Pattern Recognition 2017; 61: 539–556. doi: 10.1016/j.patcog.2016.07.001

[11] Nasrabadi NM. DeepTarget: An automatic target recognition using Deep Convolutional Neural Networks. IEEE Transactions on Aerospace and Electronic Systems 2019; 55 (6): 2687–2697. doi: 10.1109/TAES.2019.2894050

[12] Sildir H, Aydin E, Kavzoglu T. Design of Feedforward Neural Networks in the classification of hyperspectral imagery using superstructural optimization. Remote Sensing 2020; 12 (6): 956. doi: 10.3390/rs12060956

[13] Alganci U, Soydas M, Sertel E. Comparative research on deep learning approaches for airplane detection from very highresolution satellite images. Remote Sensing 2020; 12 (3): 458. doi: 10.3390/rs12030458

[14] Zhao W, Du S. Learning multiscale and deep representations for classifying remotely sensed imagery. ISPRS Journal of Photogrammetry and Remote Sensing 2016; 113: 155–165. doi: 10.1016/j.isprsjprs.2016.01.004

[15] Chen Y, Jiang H, Li C, Jia X, Ghamisi P. Deep feature extraction and classification of hyperspectral images based on Convolutional Neural Networks. IEEE Transactions on Geoscience and Remote Sensing 2016; 54 (10): 6232–6251. doi:10.1109/TGRS.2016.2584107

[16] Kavzoglu T, Teke A, Yilmaz EO. Shared blocks-based ensemble deep learning for shallow landslide susceptibility mapping. Remote Sensing 2021; 13 (23): 4776. doi: 10.3390/rs13234776

[17] Alam M, Wang JF, Guangpei C, Yunrong L, Chen Y. Convolutional Neural Network for the Semantic Segmentation of Remote Sensing Images. Mobile Networks and Applications; 2021; 26 (1): 200–215. doi: s11036-020-01703-3

[18] Chen J, Huang H, Peng J, Zhu J, Li W et al. Convolution Neural Network architecture learning for remote sensing scene classification. arXiv, 2020, arXiv:2001.09614

[19] Maggiori E, Tarabalka Y, Charpiat G, Alliez P. Convolutional Neural Networks for large-scale remote-sensing image classification. IEEE Transactions on Geoscience and Remote Sensing 2017; 55 (2): 645–657. doi: 10.1109/TGRS.2016.2612821

[20] Hamwood J, Alonso-Caneiro D, Read SA, Vincent SJ, Collins MJ. Effect of patch size and network architecture on a Convolutional Neural Network approach for automatic segmentation of OCT retinal layers. Biomedical Optics Express 2018; 9 (7): 3049. doi: 10.1364/boe.9.003049

[21] Paoletti ME, Haut JM, Plaza J, Plaza A. A new Deep Convolutional Neural Network for fast hyperspectral image classification. ISPRS Journal of Photogrammetry and Remote Sensing 2018; 145: 120–147. doi: 10.1016/j.isprsjprs.2017.11.021

[22] Martins J, Nogueira K, Zamboni P, de Oliveira PTS, Gonçalves WN et al. Segmentation of tree canopies in urban environments using Dilated Convolutional Neural Network. International Geoscience and Remote Sensing Symposium (IGARSS); 2021; 6932–6935. doi:10.1109/IGARSS47720.2021.9553218

[23] Song H, Kim Y, Kim Y. A patch-based light Convolutional Neural Network for land-cover mapping using Landsat-8 images. Remote Sensing 2019; 11 (2): 114. doi: 10.3390/rs11020114

[24] Farooq A, Hu J, Jia X. Weed classification in Hyperspectral remote sensing images via Deep Convolutional Neural Network. 018 IEEE International Geoscience and Remote Sensing Symposium; 2018: 3816–3819. doi: 10.1109/IGARSS.2018.8518541

[25] Liu B, Du S, Du S, Zhang X. Incorporating deep features into GEOBIA paradigm for remote sensing imagery classification: A patch-based approach. Remote Sensing 2020; 12 (18): 3007. doi: 10.3390/rs12183007

[26] Mosavi A, Ardabili S, Várkonyi-Kóczy AR. List of Deep Learning Models. In: Várkonyi-Kóczy AR. (editor) Engineering for Sustainable Future. INTER-ACADEMIA 2019. Lecture Notes in Networks and Systems, Springer Cham, 2020, pp. 202–214.

[27] Lecun Y, Bengio Y, Hinton G. Deep learning. Nature 2015; 521(7553): 436–444. doi: 10.1038/nature14539

[28] Vali A, Comai S, Matteucci M. Deep learning for land use and land cover classification based on hyperspectral and multispectral Earth observation data: A Review. Remote Sensing 2020; 12 (15): 2495. doi: 10.3390/rs12152495

[29] Khan N, Shah J, Stavness I. Bridgeout: Stochastic Bridge Regularization for Deep Neural Networks. IEEE Access 2018; 6: 42961–42970. doi: 10.1109/ACCESS.2018.2863606

[30] Khan A, Sohail A, Zahoora U, Qureshi AS. A survey of the recent architectures of Deep Convolutional Neural Networks. Artificial Intelligence Review 2020; 53 (8): 5455–5516. doi: 10.1007/s10462-020-09825-6

[31] Doegar A, Hiriyannaiah S, Matt SG, Gopaliyengar SK, Dutta M. Image forgery detection based on fusion of lightweight deep learning models. Turkish Journal of Electrical Engineering & Computer Sciences 2021; 29 (2021): 1978-1993. doi:10.3906/elk-2005-37

[32] Huang Y, Yu Y. An internal covariate shift bounding algorithm for Deep Neural Networks by unitizing layers' outputs. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2020; 8462–8470. doi: 10.1109/CVPR42600.2020.00849

[33] Ghassemi M, Ghassemian H, Imani M. Hyperspectral image classification by optimizing Convolutional Neural Networks based on information theory and 3D-Gabor filters. International Journal of Remote Sensing 2021; 42 (11): 4383–4413. doi:10.1080/01431161.2021.1892854

[34] Kiranyaz S, Avci O, Abdeljaber O, Ince T, Gabbouj M et al. 1D Convolutional Neural Networks and applications: A survey. Mechanical Systems and Signal Processing 2021; 151: 107398. doi: 10.1016/j.ymssp.2020.107398

[35] Liu Y, Wu L. High performance geological disaster recognition using Deep Learning. Procedia Computer Science 2018;139: 529–536. doi: 10.1016/j.procs.2018.10.237

[36] Zhang Q, Yang LT, Chen Z, Li P. A survey on Deep Learning for big data. Information Fusion 2018; 42: 146–157. doi: 10.1016/j.inffus.2017.10.006

[37] Lu H, Wang H, Zhang Q, Yoon SW, Won D. A 3D Convolutional Neural Network for volumetric image semantic segmentation. Procedia Manufacturing 2019; 39: 422–428. doi: 10.1016/j.promfg.2020.01.386

[38] Mossa AA, Cevik U. Ensemble learning of multiview CNN models for survival time prediction of brain tumor patients using multimodal MRI scans.Turkish Journal of Electrical Engineering & Computer Sciences 2021; 29 (2021): 616-631. doi:10.3906/elk-2002-175

[39] Samat A, Li E, Du P, Liu S, Miao Z. Improving Deep Forest via Patch-Based Pooling, Morphological Profiling, and Pseudo Labeling for Remote Sensing Image Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 2021; 14: 9334–9349. doi:10.1109/JSTARS.2021.3110994

[40] Li H, Zhang C, Zhang Y, Zhang S, Ding X et al. A Scale Sequence Object-based Convolutional Neural Network (SS-OCNN) for crop classification from fine spatial resolution remotely sensed imagery. International Journal of Digital Earth 2021; 14 (11): 1528–1546. doi:10.1080/17538947.2021.1950853

[41] Kavzoglu T, Colkesen I. Entropic distance based K-star algorithm for remote sensing image classification. Fresenius Environmental Bulletin 2011; 20 (5): 1200–1207.

[42] Kavzoglu T, Colkesen I. An assessment of the effectiveness of a rotation forest ensemble for land-use and land-cover mapping. International Journal of Remote Sensing 2013; 34 (12): 4224–4241. doi: 10.1080/01431161.2013.774099