Performance Comparison of Learning Methods for Soil Parameter Estimation using Hyperspectral Data

2022 8th International Conference on Signal Processing and Communication (ICSC) | 978-1-6654-5430-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICSC56524.2022.10009630

Gautham Bolar
Department of Electronics and Communication Engineering
PES University
Bengaluru 560085, India
bgautham4@gmail.com

Dheemanth Joshi
Department of Electronics and Communication Engineering
PES University
Bengaluru 560085, India
dheerj188@gmail.com

Sai Pranay Chennamsetti
Department of Electronics and Communication Engineering
PES University
Bengaluru 560085, India
chennamsetti.pranay@gmail.com

Sanjeev Gurugopinath
Department of Electronics and Communication Engineering
PES University
Bengaluru 560085, India
sanjeevg@pes.edu

***Abstract*—Estimation of soil parameters using hyperspectral imaging data is an important problem in many applications. In this work, we present a comparative performance study of three machine learning-based algorithms using linear regression, convolutional neural networks (CNN) and multilayer perceptron architectures, for the estimation problem at hand. The four parameters to be estimated include potassium, phosphorous pentoxide, magnesium and pH values. Using the dataset that was provided as a grand challenge at ICIP 2022, we show that the CNN model offers the best performance among all the other considered architectures, in terms of the Score performance metric. Moreover, all the considered three architectures outperform the baseline.**

Index Terms*—*Convolutional neural networks, hyperspectral imaging, linear regression, machine learning, perceptron.

I. Introduction

Information of the soil parameters are crucial for several applications, e.g., agriculture [1], military [2], surveillance [3] and several other applications. Due to the advancements in the fields of earth observation [4] and artificial intelligence (AI), real-time hyperspectal data can be acquired through satellite imaging and processed through AI techniques [5]. Usage of AI techniques helps in mitigating the problems such as high dimensionality and information redundancy.

The idea behind hyperspectral satellite imaging (HSI) is to analyze a wide spectrum of light, instead of mere assignment of primary colors to each pixel [6]. In HSI, images are captured across continuous wavelengths of the same spatial area with a special camera installed on an earth observing satellite [7]. In particular, each pixel in HSI carries the spectral information – which is reflectance for the problem of soil parameter estimation. This spectral information is added to the two dimensional spatial image [8] which generates a three dimensional cube known as the *hyperspectral data cube*. Compounds and parameters such as nitrogen, organic matter, potassium, magnesium, pH and moisture content are among the few that must be continuously monitored for various applications involving soil parameter estimation. In the manual soil testing, samples are collected from multiple field locations and analyzed in laboratories. This process is not only time-consuming and expensive, but also unreliable because soil parameters actively change over time. With the availability of real time hyperspectral data, this problem has become relatively easy to solve.

Several solutions using machine learning (ML) have been studied in the literature for analyzing soils using the acquired data. At a high-level, the idea is to to use hyperspectral data cube to train an ML or an AI model to estimate the soil parameters [5]. A few papers have addressed this soil parameter estimation problem, such as the urea estimation problem addressed in [9]. In a work similar to this, hyperspectral data is used to analyze heavy metals present in a given land in [5]. Methods for detection of fertilizer quantity [10] and organic matter estimation [11] in soil using HSI have also been proposed. However, most of these models relied on a handheld hyperspectral camera which covered a relatively small area, and thus make use of a small amount of data. Moreover, most of the existing solutions predict one particular soil parameter, using a single ML model. To overcome these limitations, we propose three learning models for estimating four important soil parameters using hyperspectral data of a varying area size captured. In particular, we propose the following architectures: a linear regressor (LR)-based architecture, multilayer perceptron architecture and a convolutional neural network (CNN) architecture. A comparative study of these proposed models upon varying their internal structure and parameters has been conducted to identify the best model to perform hyperspectral image processing and analysis. We show that the CNN architecture outpeforms the other architectures. The main contributions of our paper are summarized as follows. We propose three learning-based architectures for the problem of soil parameter estimation, using the linear regressor, multilayer perceptron and the convolutional neural network algorithms.

978-1-6654-5430-8/22/$31.00 ©2022 IEEE

Next, we study and compare the performances of the above mentioned models, in terms of the *Score* performance metric and show that the CNN architecture offers the best Score performance among all the proposed architectures, and also outperforms the baseline performance.

The remainder of this paper is organized as follows. Section. II presents the details on the dataset used to train and validate our architectures. Details on the proposed architectures and the corresponding scores obtained for the models are discussed in Section III. Section IV discusses the experimental results which were obtained by feeding in the data to the networks, and insights on the comparison among these models. Section V concludes the paper.

II. Dataset

The dataset used in this paper was provided as a part of the grand challenge titled *seeing beyond the visible: estimating soil parameters using hyperspectral images*, at the *international conference on image processing (ICIP 2022)* [12] [13]. The dataset was obtained by collecting an extensive set of soil samples from several agricultural fields across Poland using a hyperspectral camera mounted on an aircraft. The training data consists of 1732 training images, where each image consists of 150 contiguous hyperspectral bands $(462-942$ nm) with a spectral resolution of 3.2 nm and a ground sampling distance (GSD) of 2 m. The corresponding ground truth table (GTT) was provided for these images. The objective of the problem at hand is to estimate the selected soil parameters – specifically potassium (K), phosphorus pentoxide (P_2O5), magnesium (Mg), and pH values. A test data set of 1154 images was provided to evaluate our architecture. The metric dictated by the challenge to characterize the proposed architectures are explained below. To maintain comparability between models and for a fair evaluation, the same metric is used throughout this paper. The given GTT consists the values of soil parameters present in the agricultural fields. The model will be evaluated by its *Score* which is calculated as

$$\text{Score} = \frac{\sum_{i=1}^{4}(\text{MSE}_i/\text{MSE}_i^{\text{Base}})}{4}, \tag{1}$$

where

$$\text{MSE}_i = \frac{\sum_{j=1}^{|\psi|}(P_j - \hat{P}_j)^2}{|\psi|}, \tag{2}$$

$$\text{MSE}_i^{\text{Base}} = \frac{\sum_{j=1}^{|\psi|}(\bar{P}_i - \hat{P}_j)^2}{|\psi|}, \tag{3}$$

with $|\psi|$ representing the cardinality of the test set, and $\bar{P}_i$ represents the mean value of the i^{th} soil parameter from the training set. This implies a better model will have a lower score, zero being the perfect score.

Each hyperspectral image consists on average of 60×60 pixels in 150 bands, such that each data cube representing an image has dimensions of $60 \times 60 \times 150$ on an average [14]. Each *page* of the cube represents a particular band of the hyperspectral image, and each pixel in that particular band represents a reflectance value. Figure 1 represents the average reflectance values for image 1342, and Fig. 2 represents a page of the hyperspectral cube for image 1221. For example, a training image has dimensions $116 \times 108 \times 150$. The value stored at position $(10, 30, 111)$ represents the reflectance value of the 300^{th} pixel in the 111^{th} band [15].

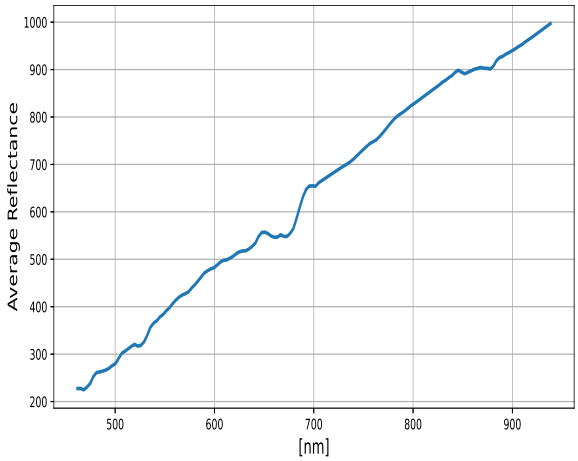

Fig. 1. An example of spectral curve for image number 1342.

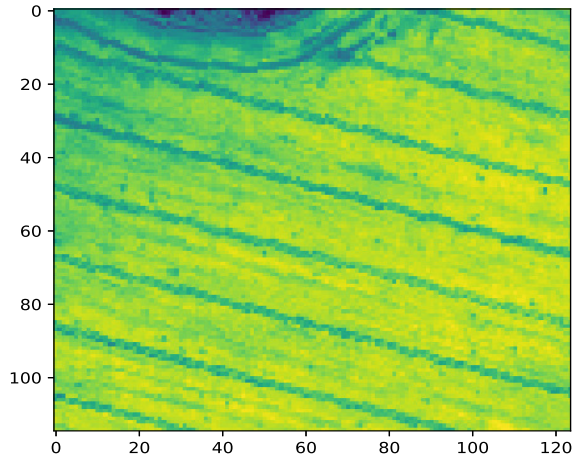

Fig. 2. Representation of band 100 (778.54 nm) of image 1221.

The GTT consists of four columns representing the four soil parameters for each training example. Thus, since there are 1732 training images, the GTT contains 1732 four point vectors, each representing one soil parameter of a particular agricultural field. A baseline algorithm was provided by the challenge organizers. This algorithm works by taking the mean values of the soil parameters obtained by a baseline regressor for the training set. Further, it generates its predictions, which is distributed as a uniform random variable over the range (x_i - $0.05 \times x_i$ to x_i + $0.05 \times x_i$) where x_i is the i^{th} soil parameter's mean value. Based on evaluation metric, the baseline algorithm obtained a Score of 1.165. This indicates that if the model randomly guesses the result, the score obtained will be higher or equal to 1.165. Therefore, the models proposed in this paper

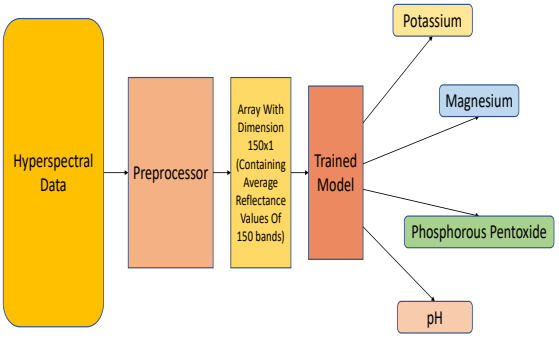

Fig. 3. An overview of the system model.

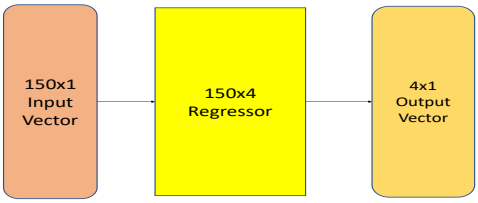

Fig. 4. Block diagram of the LR architecture.

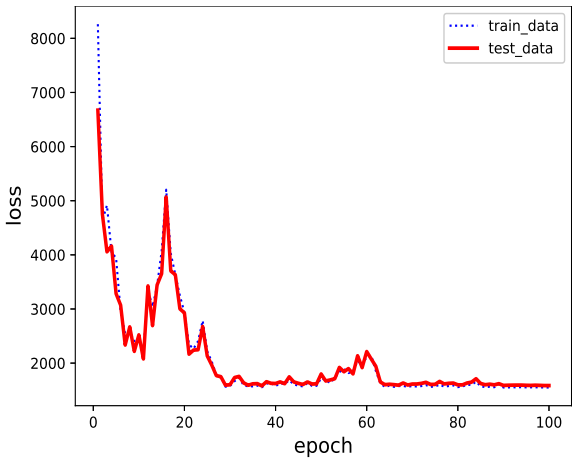

Fig. 5. Loss Vs. epoch for the LR architecture. Here, "Test data" refers to validation set.

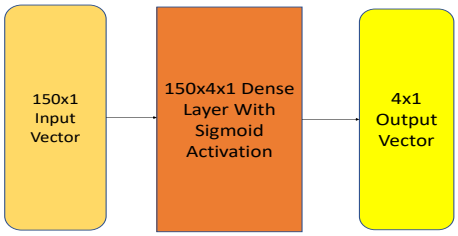

Fig. 6. Block diagram of the perceptron architecture.

intend to lower this score.

III. System Model and Proposed Architectures

As part of the competition, the GTT for the test data was not provided. In order to test the functionality of the models, 1500 images from the training data were reserved for training, and the remaining 232 images were reserved for validating the models. The final score generated and obtained is with respect to the actual GTT of the test data. Since the training data size was large, the input images were randomly shuffled to provide the proposed network with more "variance" in training. An overview of the proposed models is depicted in Fig. 3. In the following, we propose and discuss the learning architectures considered in this paper.

A. Linear Regressor Architecture

This model consists of 150 input neurons and 4 output neurons with no hidden layers, as depicted in Fig. 4. Input to the network consists of the average reflectance of each band of an image and the output represents the soil parameters. These 150 inputs are fed in to the network. Gradient calculations and optimizations are performed using stochastic gradient descent (SGD) optimizer while loss is computed using mean squared error (MSE) metric. The network is trained for 100 epochs, with a mini batch size of 50. As the number of epoch increases, loss decreases but starts to fluctuate. To stabilize this loss, we define the learning rate parameterized as a function of epoch as:

$$\text{Learning Rate} = 0.1 \times \left(1.00001 - \frac{\text{Epoch}}{\text{Num. of Epochs}}\right). \quad (4)$$

Figure 5 depicts the loss versus epoch relation of the LR architecture. A weight decay factor $\lambda = 0.001$ is used to reduce over-fitting of the model to the training set [16]. Predictions were generated with the trained LR and the following observations were made:

- When tested against the 232 training images which were not used to train the network, an average score of 0.982 was obtained. This image set is shown in Fig. 5 as the "validation set".
- A score of 0.988 was obtained by the network when tested against actual test data. The network was trained with 1500 images, excluding the validation set.

B. Perceptron Architecture and Score Values

The perceptron architecture has 150 neurons in its input layer and 4 output neurons, as depicted in Fig. 6. To optimize the network better [17], a hidden layer is added so that the network learns the trends better. This hidden layer contains 100 neurons followed by a dense output layer. This model

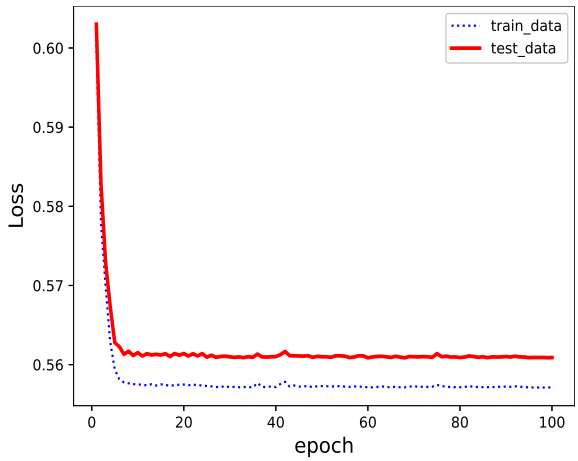

Fig. 7. Loss Vs. epoch for the perceptron architecture. Here, "Test data" refers to validation set.

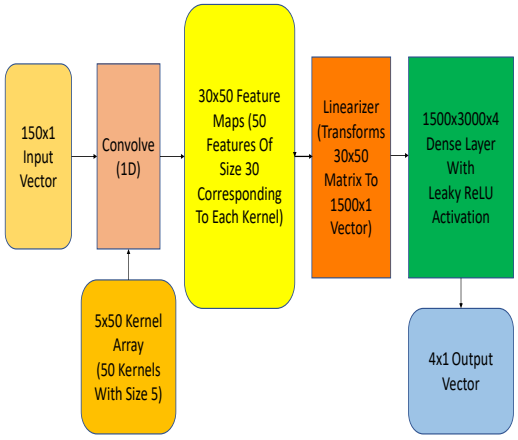

Fig. 8. Block diagram of the CNN architecture.

is trained using the SGD optimizer and loss is computed by using the binary cross entropy (BCE) defined as:

$$C = -\frac{1}{n}\sum_{x}[y\ln(a) + (1-y)\ln(1-a)], \tag{5}$$

where n denotes the output size, a denotes the model output, and y denotes the target output. Additionally, the inputs to the perceptron is normalized such that they have unit mean and unit variance. This is established to improve the learning process of the model, using the following simple formula

$$x_{\text{new}} = \frac{x - \bar{x}}{\sigma}, \tag{6}$$

where $\bar{x}$ and σ denote the sample mean and variances of x, respectively.

The learning rate used is the same as with the LR architecture. When tested against different test datasets, the following observations were made:

- An average score of 0.991 was obtained when tested against the validation set, while the network is trained with 1500 images.
- When tested against the actual test data, the network scored 0.997.

C. Convolutional Neural Network (CNN) Architecture

The input to the CNN architecture is the same as that for LR and the perceptron, and the details are depicted in Fig. 8. Fifty kernels of size $[5 \times 1]$ are introduced to capture the trends of the 4 soil parameters. The kernels are convolved independently with the input field, resulting in 50 features for one image. These features are then fed into a deep neural network to obtain the output. The loss was computed using the MSE method and the network was trained using the SGD algorithm. Leaky rectified linear unit (ReLU) is used as a nonlinear activation function for the deep neural network, which is defined as [18]

$$\text{Leaky ReLU}(x) = \begin{cases} x, & x \geq 0, \\ 0.01x, & \text{otherwise.} \end{cases} \tag{7}$$

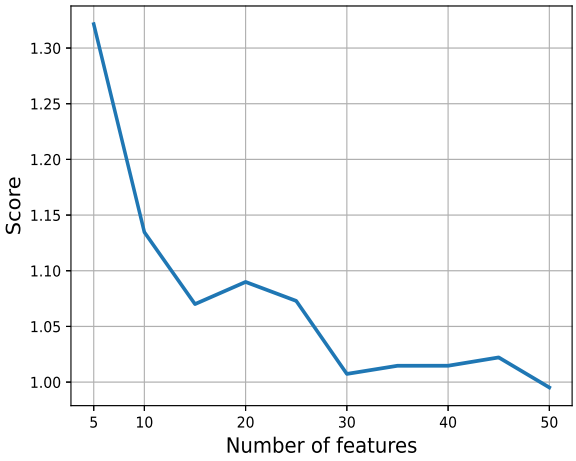

Fig. 9. Score Vs. number of features.

Plots shown in Figs. 9 and 10 were generated, which determine the scores for various features and number of hidden neurons in the fully connected layers. It is observed from Figs. 9 and 10 that a hidden layer of 3000 neurons and 50 features were the optimum pair to yield the best score for this model.

When the trained network was tested on our test sets we observed the following:

- This model resulted a average score of 0.968 when trained with 1500 training images and tested against the validation set.
- This model scored 0.981 when tested against the actual test data.

In the next section, we present the main results that were obtained from the proposed architectures and discuss them in detail.

IV. Results and Discussion

Following the discussions from Section III, and from the compiled score values given in Table I, it is observed that the CNN yields the best performance among all the other proposed architectures. Even though the multilayer perceptron

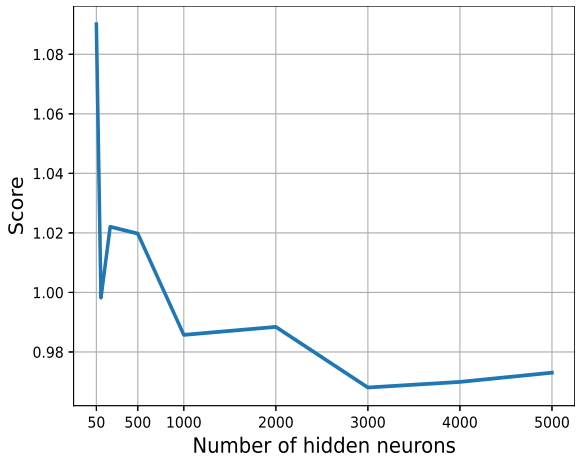

Fig. 10. Score Vs. number of hidden neurons.

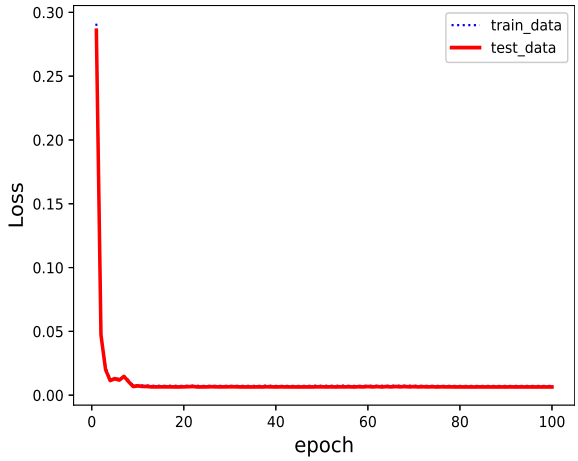

Fig. 11. Loss Vs. epoch for the perceptron architecture. Here, "Test data" refers to validation set.

(which predicts the output based on a weighted sum of inputs) is expected to perform better than LR, it still yields in inferior performance. This may be due to the fact that the perceptron intends to capture variations in the data by judging the variations across the full range (150 bands), which could make it more susceptible to variations present in the input images. Note that the CNN architecture overcomes this problem by weighing the influence of soil parameters across the neighbourhood of a certain band, unlike the other models. As seen in Fig. 11, this model also solves the problem of over-fitting. Performance of these models are noted to substantially change with respect to their parameters, such as number of hidden layers, number of neurons in a hidden layer, etc. In the following, we will discuss about the variations in scores when the parameters mentioned above are varied in individual models.

TABLE I
BEST *scores* OBTAINED FROM PROPOSED MODELS. BASELINE SOLUTION SCORE OBTAINED FROM [13]. THE BASELINE MODEL WAS NOT TESTED AGAINST THE TEST DATA.

Model	Validation score	Test score
Baseline [13]	1.165	Not Available
Perceptron	0.991	0.997
Linear regressor	0.982	0.988
Convolutional neural network	0.968	0.981

LR Architecture: The LR architecture has no hidden layers, and hence there is no "practical architecture" that can be changed in this model. Mathematically, this model creates a linear hyperplane in a 150 dimensional space, that classifies/estimates the amount of soil parameters, and provides a four dimensional output.

Perceptron Architecture: The main idea of a deep neural network built with perceptron algorithm is to approximate a function by learning the function [19]. If the function to be learned is complex, more number of hidden layers are added to learn the function better. However in this problem, when a neural network with 2 or more hidden layers is trained and tested, the predictions are saturating to a certain value which is close to the mean. This problem might be due to the *gradient decay* problem. This means that the weights and biases of the early hidden layers may not learn at a rate at which the later hidden layers are learning, resulting in inefficient optimization of the variables and estimating a near same value around the mean. Overfitting is an another drawback of this model [20]. In Fig. 7, it can be seen that there is a significant difference in loss saturation levels between training and test (validation set) data when the system is trained with the perceptron algorithm. This suggests that the model's ability to *generalize* is poor.

CNN Architecture: When compared with other two models, the CNN architecture is most susceptible when the architecture parameters are varied. CNN model weighs the bands based on the information stored in each bands unlike the perceptron algorithm where each band is given the same weight. This means that during training, the kernels are optimized such that feature maps produced contain weighted information on the important bands and less weighted information on unimportant bands [21]. This basically helps the model to classify the noisy bands and the weighed bands [22] [23]. This makes the regression process more efficient. The efficiency of the architecture is further increased by adding more neurons to the hidden layer of the regressor. Adding more neurons to the regressor approximates the function better, which helps the model to estimate the outputs more accurately, this can be observed in Fig. 10. Adding additional features (channels) also improves the score, as seen in Fig. 9. The CNN model is sensitive when changes are made in the architecture unlike the perceptron model where the estimations saturate to a value close to the mean due the gradient decay problem. As can be seen in the Fig. 1, there is a dip between 637.89 nm and 701.82 nm that relates to bands in the (56.76) region. This indentation occurs consistently in most of the images in the training data set. When the perceptron algorithm was trained with only these bands, it was observed that the result improved with a small margin. When comparing the individual results, the Score for magnesium improved significantly, suggesting that the light

absorbed in the band region may be due to magnesium.

V. Conclusion and Future Work

We considered the soil parameter estimation problem using earth observation, which mainly dealt with hyperspectral data that yields more information than a normal color image. Specifically, we considered three ML models namely LR, CNN and perceptron architectures, to learn the trends of the given data (reflectance values). The corresponding scores were evaluated and compared. It was observed that the CNN model resulted in the best score among the three models, and was observed to be more susceptible towards the variations in the architecture of the hidden layers. This problem can be further explored by introducing other stronger architectures with different supervised algorithms using techniques such as radial basis functions and long short-term memory. Other metrics such as the ℓ_1 function can be incorporated to characterize the performance of this estimation problem.

References

[1] U. Seiffert, F. Bollenbeck, H.-P. Mock, and A. Matros, "Clustering of crop phenotypes by means of hyperspectral signatures using artificial neural networks," in *Proc. 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2010, pp. 1–4.

[2] Y. Montembeault, P. Lagueux, V. Farley, A. Villemaire, and K. C. Gross, "Hyper-cam: Hyperspectral IR imaging applications in defence innovative research," in *Proc. 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2010, pp. 1–4.

[3] A. Koz, "Ground-based hyperspectral image surveillance systems for explosive detection: Part I—state of the art and challenges," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 12, pp. 4746–4753, 2019.

[4] M. Borgeaud, M. Drinkwater, P. Silvestrin, and M. Rast, "Status of the ESA earth explorer missions and the new ESA earth observation science strategy," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2015, pp. 4189–4192.

[5] Y. Zhang, Y. Xu, W. Xiong, R. Qu, J. Ten, Q. Lou, and N. Lv, "Inversion study of heavy metals in soils of potentially polluted sites based on UAV hyperspectral data and machine learning algorithms," in *Proc. 11th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2021, pp. 1–5.

[6] X. Yang, Y. Ye, X. Li, R. Y. K. Lau, X. Zhang, and X. Huang, "Hyperspectral image classification with deep learning models," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 9, pp. 5408–5423, 2018.

[7] J. Transon, R. d'Andrimont, A. Maugnard, and P. Defourny, "Survey of current hyperspectral earth observation applications from space and synergies with sentinel-2," in *Proc. 9th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MultiTemp)*, 2017, pp. 1–8.

[8] R. Sugiarto, A. H. Saputro, and W. Handayani, "Study on venation visualization system using hyperspectral images and multi-layer perceptron classifier," in *Proc. International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 2019, pp. 478–482.

[9] A. K. Patel, J. K. Ghosh, S. Pande, and S. U. Sayyad, "Deep-learning-based approach for estimation of fractional abundance of nitrogen in soil from hyperspectral data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 6495–6511, 2020.

[10] J. P. Kumar, S. Deshpande, and A. Inamdar, "Detection of fertilizer quantity in soil using hyperspectral data," in *Proc. 9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, 2018, pp. 1–4.

[11] Z. Gong, Q. Wang, C. Zhang, and H. Guan, "Soil organic matter estimation using hyperspectral remote sensing techniques in a water-level-fluctuating zone around guanting reservoir, Beijing, China," in *Proc. IEEE International Geoscience and Remote Sensing Symposium(IGARSS)*, 2019, pp. 6929–6932.

[12] "IEEE International Conference on Image Processing." [Online]. Available: https://2022.ieeeicip.org/

[13] "Seeing beyond the visible." [Online]. Available: https://platform.ai4eo.eu/seeing-beyond-the-visible

[14] D. Varade, A. Sure, and O. Dikshit, "Evaluation of multi-spectral cube from multi-sensor imagery corresponding to hyperspectral imagery," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2017, pp. 656–659.

[15] K. Shang, C. Xiao, H. Wei, and Y. Xie, "Selection strategy of classification methods for hyperspectral remote sensing data," in *Proc. 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA)*, 2016, pp. 319–323.

[16] K. Nakamura and B.-W. Hong, "Adaptive weight decay for deep neural networks," *IEEE Access*, vol. 7, pp. 118 857–118 865, 2019.

[17] F. Khajehrayeni and H. Ghassemian, "Improving performance of hyperspectral unmixing using multi-layer perceptron by generating an appropriate synthetic library," in *Proc. 27th Iranian Conference on Electrical Engineering (ICEE)*, 2019, pp. 1303–1308.

[18] J. Xu, Z. Li, B. Du, M. Zhang, and J. Liu, "Reluplex made more practical: Leaky ReLu," in *Proc. IEEE Symposium on Computers and Communications (ISCC)*, 2020, pp. 1–7.

[19] K. Tarun, N. V. Reddy, T. Akhilesh, and P. Sathish, "Analysis of Doppler collision prediction using supervised machine learning," in *Proc. 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2021, pp. 973–976.

[20] H. Li, J. Li, X. Guan, B. Liang, Y. Lai, and X. Luo, "Research on overfitting of deep learning," in *Proc. 15th International Conference on Computational Intelligence and Security (CIS)*, 2019, pp. 78–81.

[21] J. Nalepa, L. Tulczyjew, M. Myller, and M. Kawulok, "Hyperspectral image classification using spectral-spatial convolutional neural networks," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2020, pp. 866–869.

[22] J. Nalepa and M. Stanek, "Segmenting hyperspectral images using spectral convolutional neural networks in the presence of noise," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2020, pp. 870–873.

[23] P. R. Lorenzo, L. Tulczyjew, M. Marcinkiewicz, and J. Nalepa, "Hyperspectral band selection using attention-based convolutional neural networks," *IEEE Access*, vol. 8, pp. 42 384–42 403, 2020.