Retraction

Retracted: An Efficient Multispectral Image Classification and Optimization Using Remote Sensing Data

Journal of Sensors

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] S. Janarthanan, T. Ganesh Kumar, S. Janakiraman, R. K. Dhanaraj, and M. A. Shah, "An Efficient Multispectral Image Classification and Optimization Using Remote Sensing Data," *Journal of Sensors*, vol. 2022, Article ID 2004716, 11 pages, 2022.

Research Article

An Efficient Multispectral Image Classification and Optimization Using Remote Sensing Data

S. Janarthanan [ID],[1] **T. Ganesh Kumar** [ID],[1] **S. Janakiraman** [ID],[2] **Rajesh Kumar Dhanaraj** [ID],[1] **and Mohd Asif Shah** [ID][3]

[1]*School of Computing Science and Engineering, Galgotias University, India*
[2]*Pondicherry University, Puducherry, India*
[3]*Kebri Dehar University, Ethiopia*

Correspondence should be addressed to Mohd Asif Shah; ayanlehassen@kdu.edu.et

A significant amount of effort and cost is required to collect training samples for remote sensing image classifications. The study of remote sensing and how to read multispectral images is becoming more important. High-dimensional multispectral images are created by the various bands that show how materials behave. The need for more information about things and the improvement of sensor resolutions have led to the creation of multispectral data with a higher size. In recent years, it has been shown that the high dimensionality of these data makes it hard to preprocess them in multiple ways. Recent research has demonstrated that one of the most crucial methods to address this issue is by adopting a variety of learning strategies. But as the data gets more complicated, these methodologies are not adequate to support. The proposed methodology shows that the classification experiment using remote sensing images indicates the maximum likelihood classifier with different deep learning models; weight vector (WV) AdaBoost and ADAM can greatly limit overfitting, and it obtains high classification accuracy. Proposed VGG16 and Inception v3 increase classification accuracy along with optimization process produce 96.08%.

1. Introduction

Multispectral sensors now capture the earth's surface reflectance in hundreds of frequency bands because of the advancement in sensor technology. As a result, multispectral pictures may be used for a wide range of activities, from categorization to environmental monitoring. For example, classification accuracy is reduced when dimensionality grows and training samples are restricted, according to the Hughes phenomenon. Multispectral images may contain strong correlations between [1] neighboring and nonadjacent bands, which lowers the quantity of data that may be used for further analysis, including categorization. Accurate categorization is dependent on the extraction of certain characteristics. Band Correlation Clustering (BCC) is a new unsupervised feature extraction [2, 3] approach introduced in this paper. There are three key processes in the suggested technique, which are the bands' correlation coefficient is cal-

culated, the bands are clustered according to the correlation coefficient matrix, and the means of each cluster are calculated with a new methodology for feature extraction. Classification accuracy and time consumption are used to assess the support vector machine approach. [4] The resulting features are supplied into nonparametric Support Vector Machine (SVM) and parametric Machine Learning (ML), two supervised classifications, for the assessment process. [5] Comparable results are obtained using [6] unsupervised feature extraction via clustering. An evaluation of the findings demonstrates that the suggested BCC performs well in terms of computing expenses to increase the accuracy in classification steps.

Object-Based Image Classification (OBIC) [7] is used to classify Very-High-Resolution (VHR) pictures. Most OBIC [8] classification algorithms use 1D features hand-crafted from picture objects (superpixels). This letter introduces a deep OBIC framework utilizing [9, 10] Convolutional
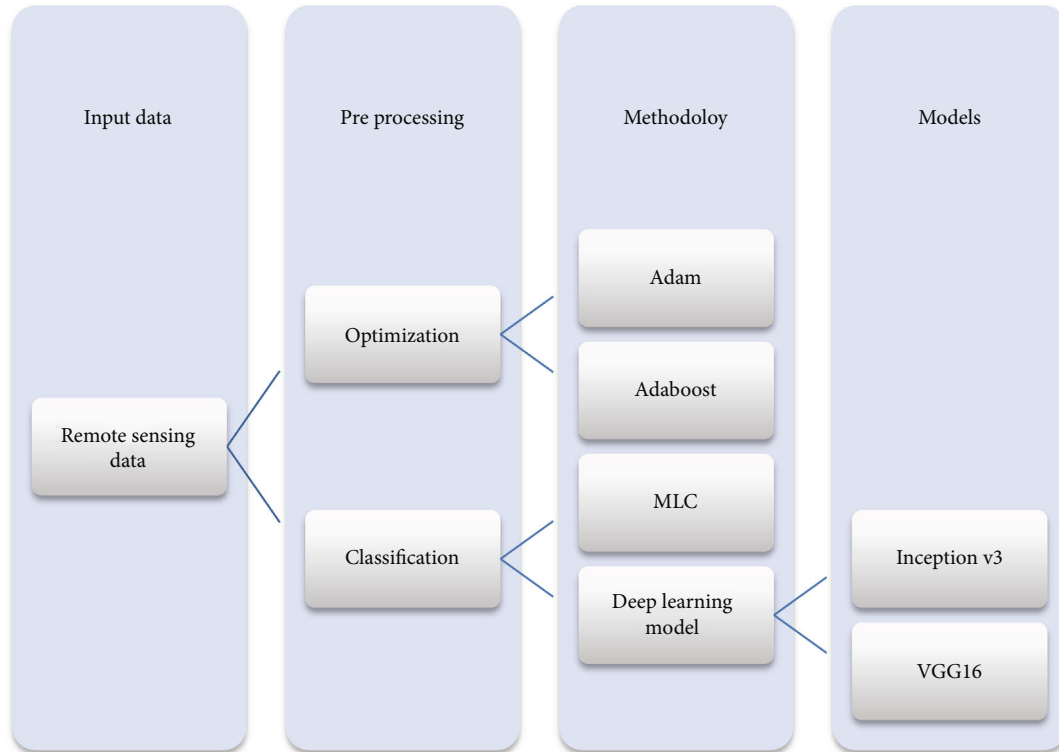
Figure 1: Classification and optimization processing steps.

Neural Networks (CNNs) to extract 2D deep superpixel features [11]. Before designing the network [12], studied superpixel mask regulations before experiments, the proposed framework for better overall accuracy, coefficient, and F-measure is delivered by our DiCNN-4 (Double-input CNN) [13] model. [14] On picture dataset than standard OBIC approaches. AdaBoost is a fantastic ensemble learning technique that combines several weak classifiers to create a strong classifier, eventually increasing the classification accuracy. [15] However, the AdaBoost combination overlooks the performance of basic classifiers at the per-class level and concentrates on their total performance. AdaBoost's ability to enhance classification accuracy is hampered by this, [16] which makes overfitting a concern in subsequent rounds. In this paper, an enhanced [17, 18] AdaBoost algorithm with Weight Vector (WV AdaBoost) is suggested to reduce these drawbacks and preserve the advantages of Ada-Boost. Each class is assigned a weight to indicate the recognition [16, 19] ability of the base classifiers using weight vectors. AdaBoost and WV AdaBoost base classifiers are trained using an Artificial Neural Network (ANN), Naive Bayes, and a decision tree [20]. The classification experiment using Remote Sensing (RS) data demonstrates that WV Ada-Boost beats AdaBoost by producing much greater classification accuracy. It can greatly reduce overfitting, within a limited number of repetitions, and WV AdaBoost may increase classification accuracy to the greatest possible level.

Labeling samples from each image is frequently a necessary step in the processing of multitemporal remotely sensed data, but it is a laborious and time-consuming operation. [21] The ground items frequently do not change consider-ably over time, thus, certain labels can be reused with the proper consistency checks. Using just one labeled picture, a new framework for weakly supervised transfer learning [22] is described in this process to categorize multitemporal remote sensing images. [23] Our system can categorize all the other multitemporal images chronologically without any labeling effort by exploiting the consistency of time-series images and a domain adaption mechanism. [24] Our system obtains a classification accuracy that is comparable to what would be obtained with [25] supervised learning. [26] With the training samples for one temporal dataset, our system is still able to handle multitemporal remote sensing images, as mentioned Figure 1 with input data, preprocessing and methodology applied, and inception v3 and VGG16 model for image classification and accuracy.

1.1. The Motivation for the Proposed Work. The main motivation for multispectral image classification and receiving better accuracy as a result compared with multistage techniques and predefined models applied to improvise the accuracy of the remote sensing data set with inception v3 and VGG16, primary goal and motivation required to analyze and improvise the accuracy with optimization techniques to increase the accuracy and computation cost reduced.

2. Related Work

2.1. Methodology. By utilizing the Adam optimizer and comparing it to VGG16 [27] and Inception v3, the suggested technique produce the result with excellent accuracy while
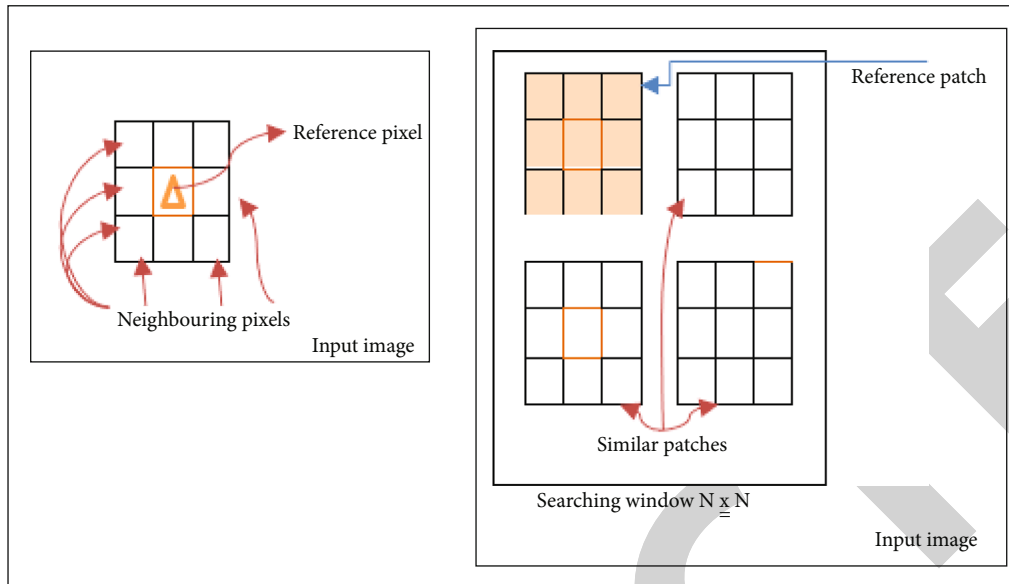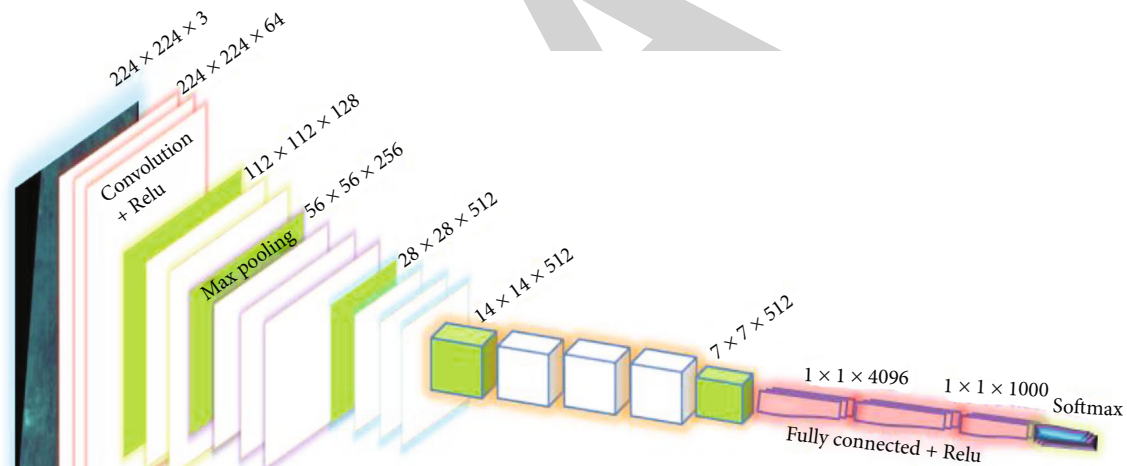
FIGURE 2: Patch extraction process.

FIGURE 3: VGG 16 model process flow architecture.

requiring minimal computing time. The following model is described in detail to train a deep learning network.

2.1.1. Normalization. This article analyzes remote sensing data using single-layer and deep convolutional networks. Given the enormous input data dimensionality and little labeled data, direct application of supervised (shallow or deep) convolutional networks to multi- and [28] hyperspectral imaging is problematic. The recommend combining unsupervised learning [29] of sparse features with greedy layer-wise unsupervised pretraining. [30] The technique uses sparse representations to enforce population and lifetime sparsity and to compute the logarithm of every pixel using data normalization.

$$a = \frac{((b - \min) * 255)}{(\max - \min)}, \tag{1}$$

where b represent the grayscale value of the original image, and a represent the grayscale value after normalized image. min and max were the grayscale level of the sample image with minimum and maximum range were 0 to 255.

2.1.2. Patch Extraction. The training image set consists of image patches from the stacked covariance matrix. [31] The approach group image patches into k clusters. Random mini-batches are retrieved with 50% overlap (stride 8) and resolution determines patch sizes. High resolution requires
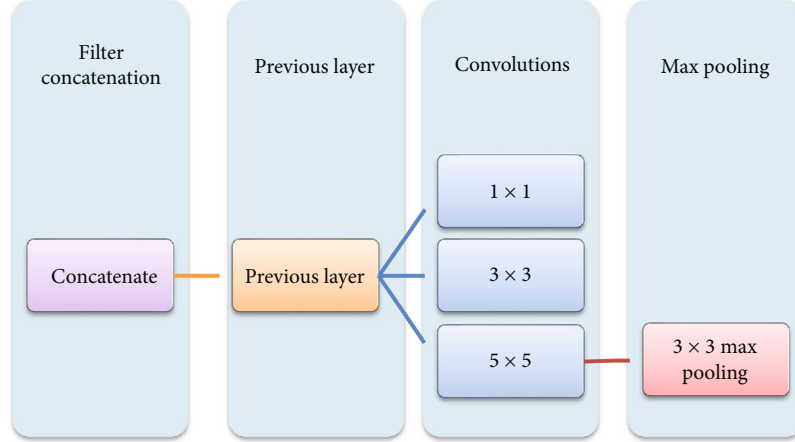
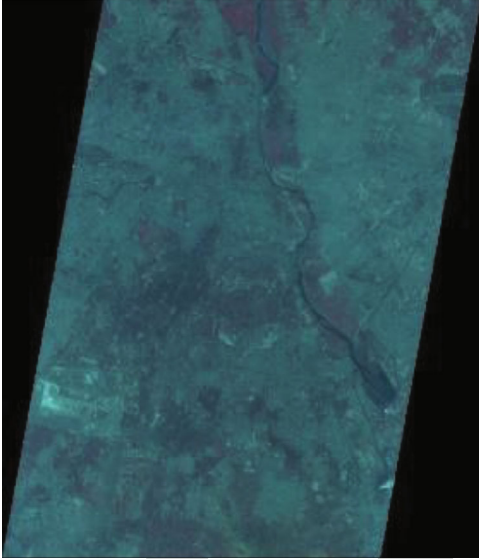FIGURE 4: Inception operation with reduced dimensions.

FIGURE 5: Combined 3 band tiff image.

tiny patches with the testing image set has 16×16 nonoverlapping picture patches. [32] These pure patches are labeled using the posterior probability computation. In Figure 2, input image with reference pixel and neighboring pixels with similar patches are explained.

2.1.3. Model Training. Batch normalization and VGG16 feature extraction are used. The classification [33] layer is fully linked, batch normalized, and ReLU. The Xavier approach [34] initializes the fully connected layer's weights. Adam uses a 0.001 learning rate and 0.00001 weight decay. This model clusters remote sensing data (GDC-SAR). In Figure 3, input image and convolution plus ReLU and max pooling resolution are reduced from $224 \times 224 \times 3$ with $1 \times 1 \times 1000$ with fully connected ReLU and softmax with VGG16 architecture.

2.1.4. Maximum Likelihood Classifier. One of the most often used techniques for categorization in remote sensing is the maximum likelihood classifier, which places a pixel into

the class that it most closely resembles. The posterior probability of a pixel belonging to class k is used to define the likelihood L_k.

$$L_K = P\left(\frac{k}{K}\right) = P(k) * \frac{P(X/k)}{\sum P(i) * P(X/i)}, \tag{2}$$

where, $P(k)$ is the prior probability of class k,.and $P(X/k)$ is the probability density function or conditional probability to witness X from class k.

$\sum P(i) * P((X/i))$ is likewise shared by all classes, and $P(k)$ is typically believed to be equal to each other as well. As a result, $P(X/k)$, or the probability density function, determines L_k. The probability density function is based on the multivariate normal distribution for mathematical reasons. The probability in the case of normal distributions may be stated as follows:

$$L_k(X) = \frac{1}{((2\pi)n)/(2|\Sigma_k|(1/2))} \exp\left| -\frac{1}{2}(X - \mu_k)\sum_k^{-1}(X - \mu_k)t \right|, \tag{3}$$

where, n the number of bands, X image data of bands n, and $L_k(X)$ belongs to the class k. μ_k is the average class k vector, Σ_k is the covariance matrices and class k variance, and $|\Sigma_k|$ is an indicator of Σ_k.The likelihood is the same as the Euclidean distance when the variance-covariance matrix is symmetric, and it is the same as the Mahalanobis distances when the determinants are equal.

2.1.5. Inception v3. A module for Google Net, Inception v3 is a convolutional neural network that aids in object recognition and image analysis. The Google Inception Convolutional Neural Network, which was first shown at the ImageNet Recognition Challenge, is in its third iteration, In Figure 4 represents the filter concatenation operation along with previous layer and convolutions layer 1x1, 3x3 and 5x5 with max pooling of 3x3.

Table 1: Parameters of IRS P-6 LISS IV Satellite.

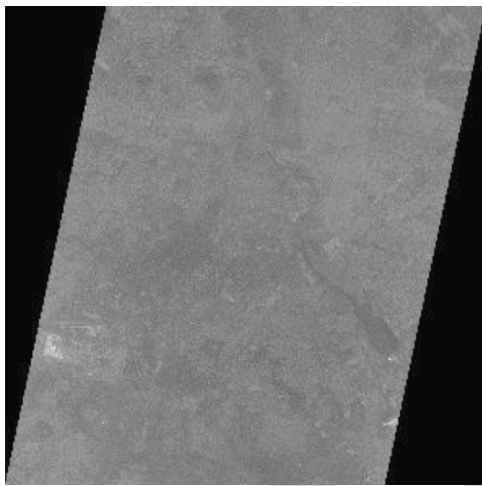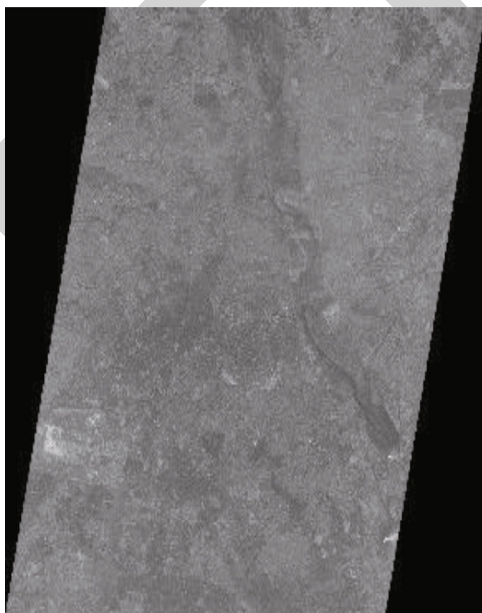| Parameter	Description
Parameter	LISS IV data
Sensor	L4MX
Sat-ID	IRS-P6
Product ID	142866521
Sensor orientation	Delhi
SAR band	3 band
Angle range	Latitude 28.615513
	Longitude 77.216714
Image format	GEOTIFF
Range resolution	5.8 m

Figure 6: Band 2 tiff image.
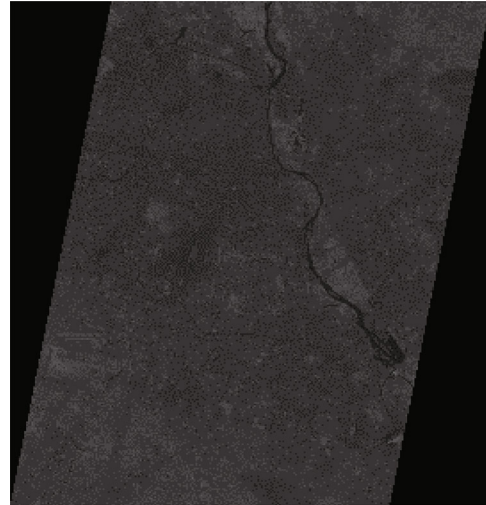

Figure 7: Band 3 tiff image.

Figure 8: Band 4 tiff image.

2.1.6. Adam Optimization Process. The Adam optimization algorithm's primary goal is quicker computing [35] with a limited number of tuning parameters such as epochs, learning rate, batch size, learning rate, optimizer, and a number of neurons. Additionally, adjust the number of layers for beta 2 second moment estimations of 0.999 and near to 1.0 and beta 1 decay rates of 0.9.

First, the gradient (g) with time (t) step.

$$g(t) = f'(X(t-1)). \qquad (4)$$

Second, to calculate moments, moving average (m) with hyperparameter beta 1.

$$m(t) = \text{beta1} * m(t-1) + (1 - \text{beta1}) * g(t). \qquad (5)$$

Then update the moments with moving average (v) with a second in beta 2.

$$v(t) = \text{beta2} * v(t-1) + (1 - \text{beta2}) * g(t). \qquad (6)$$

Next, bias adjusted with correction for the 1st moment

$$m\text{hat}(t) = \frac{(m(t))}{((1 - beta1)(t))}. \qquad (7)$$

And then 2nd moment

$$v\text{hat}(t) = \frac{v(t)}{(1 - beta2(t))}. \qquad (8)$$

A static decay schedule is required to be applied.

$$\begin{aligned} \text{beta1}(t) &= \text{beta1}^t, \\ \text{beta2}(t) &= \text{beta2}^t. \end{aligned} \qquad (9)$$

```
Read: Input data (stored in the folder)
Decode: Convert it into JPEG to RGB grid pixels with channels
Convert it into a floating point
        Input: Neural network
Rescale the pixel:
        Range: Between (0 : 255) to the [0,1]
Output: Preprocessed tensors.
```

ALGORITHM 1: Algorithm Procedure: Preprocessing.

Source image labeled: $Y_s = \{(a_i^s, b_i^s)\}_{i=1}^{n^s}$

Target image unlabeled: $Y_t = \{(a_j^t)\}_{j=1}^{n^t}$

Class labels, $Y_t = \{(b_j^t)\}_{j=1}^{n^t}$

Assign parameters:
Epoch value =100;
Min-batch size: Bs =32;
Learning rate $\alpha = 2.0\, X\, 10^{-4}$,
1st and 2nd decay rate exponential
With $\beta_1 = 0.9, \beta_2 = 0.999$ and
Epsilon: $1.0\, X\, 10^{-8}$
Initially use VGG16 and inception v3 model trained with dataset and pre-training with CNN to initialize feature generation G and classified images $C1, C2, C3$ and $C4$ alone
    for epoch = 1 : num_epoch do
Shuffle the random source and target image data set and organize it into N_b
Groups each size m
    for $k = 1 : N_b do$
Select mini-batches: $Y_{sk}, Y_{tk}\, from\, Y_s$ and Y_t
Need to train the $G, C1, C2, C3$ AND $C4\, on\, Y_{sk}$ by optimizing (1) to (10) using Adam same is required to be done for C1 to C4 (11)
End for
End for
Classify the different class target domain $\{(a_j^t)\}_{j=1}^{n^t}$ using source and target
Return $\{(b_j^t)\}_{j=1}^{n^t}$

ALGORITHM 2: Proposed Inception v3 and VGG 16 algorithm.

Finally, calculate the value of the iteration of the parameter.

$$x(t) = x(t-1) - \text{alpha} * \frac{m\text{hat}(t)}{\text{sqrt}(v\text{hat}(t)) + \text{eps}}, \qquad (10)$$

where alpha is step size and eps are small value (epsilon) and sqrt () is the square root function.

2.1.7. AdaBoost Process. Adaboost first chooses a training subset at random. By choosing the training set depending on the precision of the previous training, it iteratively trains the AdaBoost machine learning model. It gives incorrectly categorized observations at a [15] larger weight so that they will have a higher chance of being correctly classified in the upcoming round. Additionally, based on the trained classi-

fier's accuracy, weight is assigned to it in each iteration. [36] The more accurate classifier will be given more weight. This method iterates until the entire training set fits perfectly, or until the stated maximum number of estimators has been reached [37] to categorize the voting algorithm created for the selection.

3. Data Processing

3.1. Data Set. We used information from IRS P6 LISS IV remote sensing from ISRO for dataset analysis, refer to https://directory.eoportal.org/web/eoportal/satellite-missions/i/irs-p6. The area being studied is only the Delhi region of India, which is centered at latitude 0.00000 and longitude 75.00000 and scene center at latitude 28.615513 and longitude 77.216714, respectively. Figure 5 shows combined bands of RGB and Table 1 gives a brief description of the IRS P6 LISS IV data we used in our work. Figure 6

TABLE 2: Comparison between parameters existing techniques with the accuracy level.

Classification techniques used	Data set used	Software used	Accuracy assessment	Overall accuracy	Reference
Maximum likelihood technique	Landsat 8 satellite image	ERDAS imagine	Kappa statistic	82.5%	N. A. Mahmon, et al., 2015
RNN	LISS IV image	MAT lab	Kappa coefficient	87.69%	T. Vignesh, et al., 2021
SVM (support vector machines) and CNN	Google earth Landsat-8	MATLAB R2017a	Overall accuracy	70.89% 73.79%	M. Kim et al.2018
CNN	South Korea region – Google earth	MATLAB R2017a	Overall accuracy	95.7%	M. Kim et al.2018
VGG16 with Adam	Google earth and Bing maps	Python	Average accuracy	78.72	W. Teng et al., [26]
Proposed inception v3 with Adam	LISS IV	Python	Accuracy		

TABLE 3: Comparison between new parameters with proposed Scheme.

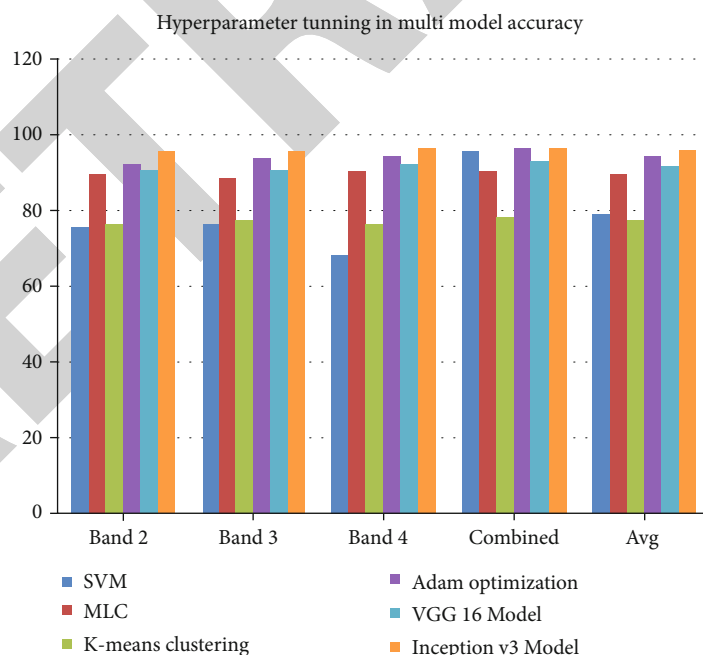| Delhi region	SVM	MLC	K-means clustering	Adam optimization	VGG 16 model	Inception v3 model
Band 2	75.68	89.52	76.72	92.21	90.80	95.50
Band 3	76.51	88.72	77.65	93.75	91.00	95.80
Band 4	68.25	90.21	76.62	94.52	92.40	96.40
Combined	95.6	90.52	78.33	96.72	93.20	96.60
Average	79.01	89.74	77.33	94.3	91.85	96.08

FIGURE 9: Comparison between existing vs. proposed multimodel accuracy.

represents band 2 tiff image, Figure 7 represents band 3 tiff image, and Figure 8 represents band 4 tiff image.

In Table 1, it mentions about Indian remote sensing p-6 LISS IV data set description in detail.

3.1.1. Data Preprocessing. The data is first level-0 preprocessed to improve quality for a more effective picture enhancement and analysis procedure. [38] Through preprocessing, binary conversion of two complex elements and
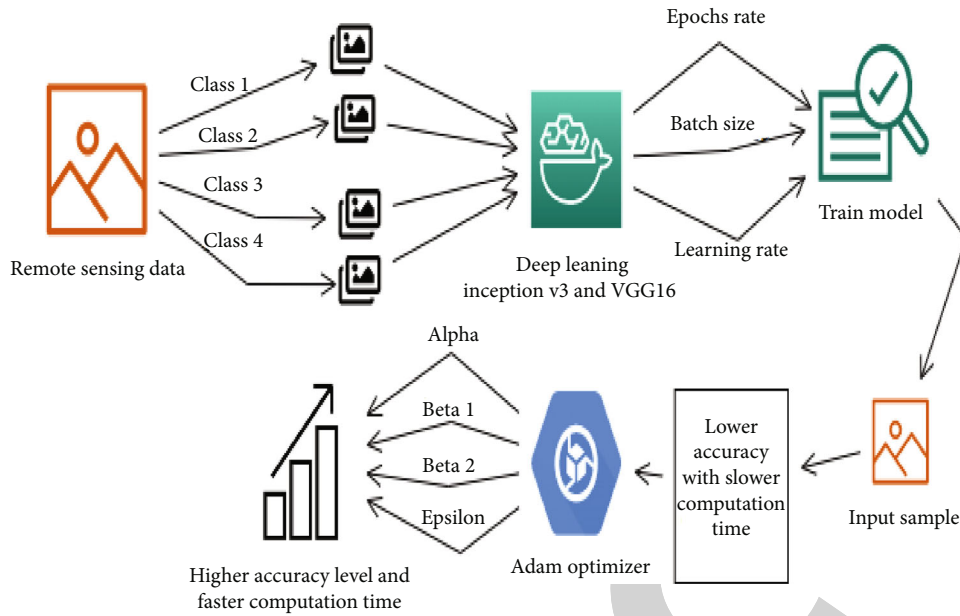
FIGURE 10: Proposed manual workflow architecture with Inception v3 and VGG16 model.

TABLE 4: User Accuracy level with sample data.

Class	Accuracy	Samples data
Class 1	1.00	1
Class 2	1.00	1
Class 3	1.00	1

these with image file formats of (.tiff extension), such that image matrix is used for patch construction and analysis, distortions are reduced and image qualities are increased. (Figures 6, 7, and 8) (Band 2, 3, and 4).

4. Results and Discussion

The three classes were used to train the model. The outcomes are also evaluated in comparison with the earlier models such as the SVM, K-means, maximum likelihood (MLC), and adaptive movement estimation. Table 2 shows the effective stochastic optimization approach with accuracy levels per class expressed as percentages.

The ultimate accuracy of a data set was assessed by averaging all of its patches. In Table 3, the range of the proposed fields of examination was advantageous. It might be challenging to distinguish mixed-class urban settlements from other urban regions. This is because there aren't many significant urban centers in these locations. The bulk of them are mixed communities, as seen in Figures 5, 6, 7, and 8. A patch may have several classes while creating the ground truth, but after categorization, it only gets one label that corresponds to the largest class. This leads to decreased accuracy along with a resolution discrepancy between SAR pictures and remote sensing data. In the above table, Adaboost is analyzed as a combined result of each band.
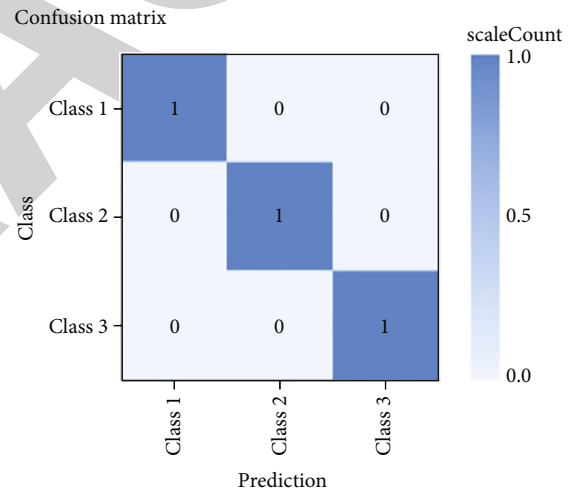

FIGURE 11: Confusion matrix.

Each model and machine learning algorithm gives the best accuracy level when compared with the new deep learning model in the python environment with ImageNet and Google Net model like VGG16 and Inception v3 model produces high accuracy level of finding the above table number. In Table 3, with the accuracy level of each band being analyzed with sample image data set for the classification accuracy train and test set prepared for the analysis of work and the manual operational categorization of sample class, the following hyperparameter tuning is used: Epochs 10, Batch size 16, learning rate 0.001, and input sample accuracy level 96.08. In Figure 9, it represents the proposed model.

The below-mentioned test results analyze with the Google AI web application with classes 1, 2, and 3, and combined bands are being analyzed with https://teachablemachine .withgoogle.com/. In Figure 10, it represents the proposed
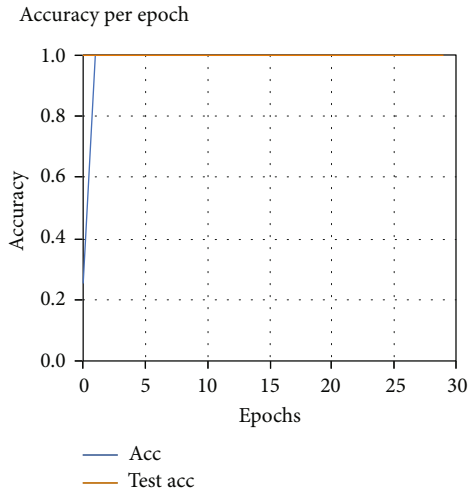
Accuracy per epoch

FIGURE 12: Accuracy per Epoch class 2.

Accuracy per epoch

FIGURE 14: Accuracy per Epoch class combined.

Accuracy per epoch

FIGURE 13: Accuracy per Epoch class 3.

Loss per epoch

x: 0
loss: 1.06928610802
test loss: 0.726546406746
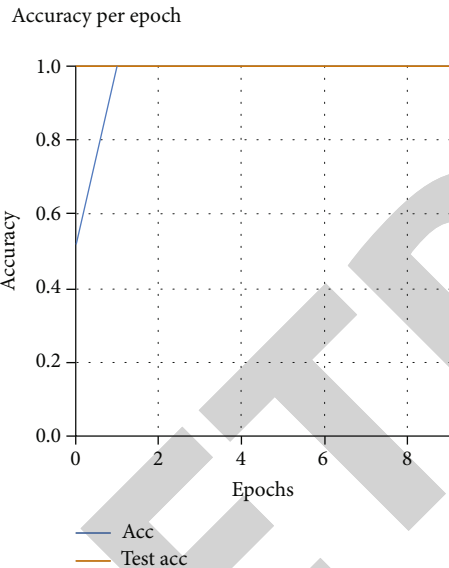
FIGURE 15: Loss per Epoch combined.

workflow with deep learning model and optimization function process clearly.

In Table 4, it represents the sample data along with various labeled sample classes were imported for the training and different hyperparameter tuning of values like batch size, epoch rate, and learning rate 0.0001 being trained with inception v3 and VGG16 model, by importing the input sample accuracy level and confusion matrix class and prediction rates shown in Figure 11.

In Figure 12, it represents the accuracy per epoch with testing accuracy comparison level of class 2 sample data. Figure 13 shows the accuracy per epoch class 3 sample data, and Figure 14 shows the combined data set range level showing in the graphical.

In Figure 15, it represents the loss per epoch combined test loss and loss range level showing in the graph.
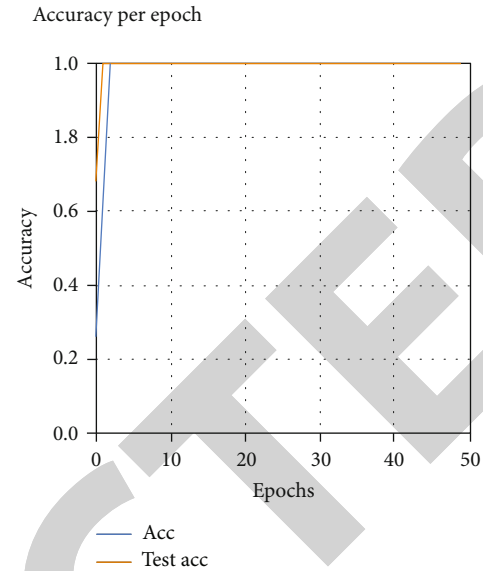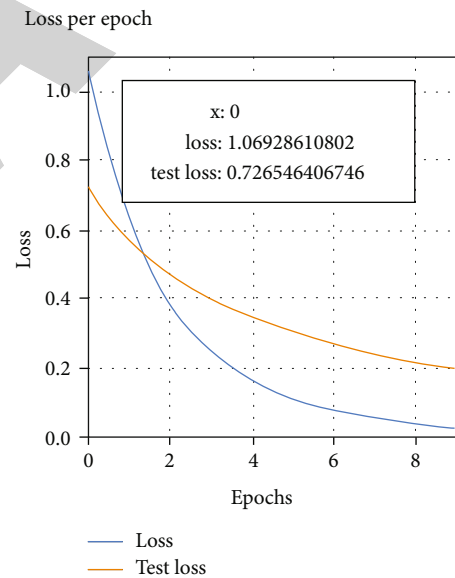
5. Conclusion and Future Scope

Deep learning is used to categorize land cover via stochastic optimization. The proposed model has a 96.08 percent accuracy rate using overlapping remote sensing picture patches. The trial findings show that the algorithm performs in line with the best practices. Hence, a general model is created, and it can be used for a variety of image data sets. Likewise, we will be able to divide the more specific classes like volume more accurately. In this research primary finding merge two deep learning model inception v3 and VGG16 with Adam optimization produce better accuracy level compare to the existing models stated in Table 2, and in the future scope

the segregation of mixed-class groups, it has not yet been investigated along with the deep learning model, and it can be considered for future research directions.

Data Availability

The data that support the findings of this study are available upon reasonable request.

Conflicts of Interest

The authors declare that they have no competing interests.

Authors' Contributions

The individual contributions of authors to the manuscript should be specified in this section. The first author analyzed and wrote the introduction, methodology, results, and the second author guided the preparation of the document with the corresponding format and support for the data set reception work area. The third and fourth authors concluded with a comparison and corresponding authors to ensure the completeness and verification procedure.

References

[1] A. Ghorbanian and A. Mohammadzadeh, "An unsupervised feature extraction method based on band correlation clustering for hyperspectral image classification using limited training samples," *Remote Sensing Letters*, vol. 9, no. 10, pp. 982–991, 2018.

[2] J. Song, S. Gao, Y. Zhu, and C. Ma, "A survey of remote sensing image classification based on CNNs," *Big Earth Data*, vol. 3, no. 3, pp. 232–254, 2019.

[3] Y. Wei, X. Luo, L. Hu, Y. Peng, and J. Feng, "An improved unsupervised representation learning generative adversarial network for remote sensing image scene classification," *Remote Sensing Letters*, vol. 11, no. 6, pp. 598–607, 2020.

[4] M. Cong, Z. Wang, Y. Tao, J. Xi, C. Ren, and M. Xu, "Unsupervised self-adaptive deep learning classification network based on the optic nerve microsaccade mechanism for unmanned aerial vehicle remote sensing image classification," *Geocarto International*, vol. 36, no. 18, pp. 2065–2084, 2021.

[5] K. A. Adepoju and S. A. Adelabu, "Improving accuracy of Landsat-8 OLI classification using image composite and multisource data with Google Earth Engine," *Remote Sensing Letters*, vol. 11, no. 2, pp. 107–116, 2020.

[6] A. Chatterjee, J. Saha, J. Mukherjee, S. Aikat, and A. Misra, "Unsupervised land cover classification of hybrid and dual-polarized images using deep convolutional neural network," *IEEE Geoscience and Remote Sensing Letters*, vol. 18, no. 6, pp. 969–973, 2021.

[7] Y. Wu, P. Zhang, J. Wu, and C. Li, "Object-oriented and deep-learning-based high-resolution mapping from large remote sensing imagery," *Canadian Journal of Remote Sensing*, vol. 47, no. 3, pp. 396–412, 2021.

[8] X. Zhang, X. Tan, G. Chen, K. Zhu, P. Liao, and T. Wang, "Object-based classification framework of remote sensing images with graph convolutional networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[9] X. Zhang, Q. Wang, G. Chen et al., "An object-based supervised classification framework for very-high-resolution remote sensing images using convolutional neural networks," *Remote Sensing Letters*, vol. 9, no. 4, pp. 373–382, 2018.

[10] S. Shi, Y. Zhong, J. Zhao, P. Lv, Y. Liu, and L. Zhang, "Land-use/land-cover change detection based on class-prior object-oriented conditional random field framework for high spatial resolution remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022.

[11] X. He and Y. Chen, "Optimized input for CNN-based hyperspectral image classification using spatial transformer network," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 12, pp. 1884–1888, 2019.

[12] R. K. Dhanaraj, V. Ramakrishnan, M. Poongodi et al., "Random forest bagging and X-means clustered antipattern detection from SQL query log for accessing secure mobiledata," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 2730246, 9 pages, 2021.

[13] X. Yu, J. Fan, J. Chen, P. Zhang, Y. Zhou, and L. Han, "Nest-Net: a multiscale convolutional neural network for remote sensing image change detection," *International Journal of Remote Sensing*, vol. 42, no. 13, pp. 4898–4921, 2021.

[14] A. E. Maxwell, T. A. Warner, and F. Fang, "Implementation of machine-learning classification in remote sensing: an applied review," *International Journal of Remote Sensing*, vol. 39, no. 9, pp. 2784–2817, 2018.

[15] P. Dou and Y. Chen, "Remote sensing imagery classification using AdaBoost with a weight vector (WV AdaBoost)," *Remote Sensing Letters*, vol. 8, no. 8, pp. 733–742, 2017.

[16] S. Janarthanan and K. Rajan, "Secure efficient geometric range queries on encrypted spatial data," 2017.

[17] S. Bera and V. K. Shrivastava, "Analysis of various optimizers on deep convolutional neural network model in the application of hyperspectral remote sensing image classification," *International Journal of Remote Sensing*, vol. 41, no. 7, pp. 2664–2683, 2020.

[18] T. R. Gadekallu, G. Srivastava, M. Liyanage et al., "Hand gesture recognition based on a Harris Hawks optimized convolution neural network," *Computers and Electrical Engineering*, vol. 100, article 107836, 2022.

[19] C. L. Chowdhary, M. Alazab, A. Chaudhary, S. Hakak, and T. R. Gadekallu, *Computer Vision and Recognition Systems Using Machine and Deep Learning Approaches: Fundamentals, Technologies and Applications*, Institution of Engineering and Technology, 2021.

[20] T. K.-A. Williams, T. Wei, and X. Zhu, "Mapping urban slum settlements using very high-resolution imagery and land boundary data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 166–177, 2020.

[21] W. Liu, R. Qin, and F. Su, "Weakly supervised classification of time-series of very high resolution remote sensing images by transfer learning," *Remote Sensing Letters*, vol. 10, no. 7, pp. 689–698, 2019.

[22] S. Krishnamoorthi, P. Jayapaul, R. K. Dhanaraj, V. Rajasekar, B. Balusamy, and S. K. Islam, "Design of pseudo-random number generator from turbulence padded chaotic map," *Nonlinear Dynamics*, vol. 104, no. 2, pp. 1627–1643, 2021.

[23] C. Guobin, Z. Sun, and L. Zhang, "Road identification algorithm for remote sensing images based on wavelet transform and recursive operator," *IEEE Access*, vol. 8, pp. 141824–141837, 2020.

[24] S. Safdar, M. Rizwan, T. R. Gadekallu et al., "Bio-imaging-based machine learning algorithm for breast cancer detection," *Diagnostics*, vol. 12, no. 5, p. 1134, 2022.

[25] R. K. Dhanaraj, K. Lalitha, S. Anitha, S. Khaitan, P. Gupta, and M. K. Goyal, "Hybrid and dynamic clustering based data aggregation and routing for wireless sensor networks," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 6, pp. 10751–10765, 2021.

[26] L. Teng, F. Xue, and Q. Bai, "Remote sensing image enhancement via edge-preserving multiscale Retinex," *IEEE Photonics Journal*, vol. 11, no. 2, pp. 1–10, 2019.

[27] K. Ramana, M. R. Kumar, K. Sreenivasulu et al., "Early prediction of lung cancers using deep saliency capsule and pre-trained deep learning frameworks," *Frontiers in Oncology*, vol. 12, 2022.

[28] H. Teffahi, H. Yao, S. Chaib, and N. Belabid, "A novel spectral-spatial classification technique for multispectral images using extended multi-attribute profiles and sparse autoencoder," *Remote Sensing Letters*, vol. 10, no. 1, pp. 30–38, 2019.

[29] S. Gadamsetty, R. Ch, A. Ch, C. Iwendi, and T. R. Gadekallu, "Hash-based deep learning approach for remote sensing satellite imagery detection," *Water*, vol. 14, no. 5, p. 707, 2022.

[30] Q. Liu, S. Basu, S. Ganguly et al., "DeepSat V2: feature augmented convolutional neural nets for satellite image classification," *Remote Sensing Letters*, vol. 11, no. 2, pp. 156–165, 2020.

[31] L. Chang, Z. Zhang, Y. Li, and X. Mao, "Multilevel extraction of vegetation type based on airborne LiDAR data," *Canadian Journal of Remote Sensing*, vol. 46, no. 6, pp. 681–694, 2020.

[32] S. Chen, W. Shi, M. Zhou, M. Zhang, and P. Chen, "Automatic building extraction via adaptive iterative segmentation with LiDAR data and high spatial resolution imagery fusion," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 2081–2095, 2020.

[33] A. M. Hilal, F. N. al-Wesabi, K. J. Alzahrani et al., "Deep transfer learning based fusion model for environmental remote sensing image classification model," *European Journal of Remote Sensing*, vol. 55, pp. 1–12, 2022.

[34] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, vol. 9, pp. 249–256, Chia Laguna Resort, Sardinia, Italy, 2010, https://proceedings.mlr.press/v9/glorot10a.html.

[35] J. Brownlee, "Code Adam optimization algorithm from scratch," *Machinelearningmastery*, 2021, https://machinelearningmastery.com/adam-optimization-from-scratch/.

[36] Z. Jiang and Y. Ma, "Accurate extraction of offshore raft aquaculture areas based on a 3D-CNN model," *International Journal of Remote Sensing*, vol. 41, no. 14, pp. 5457–5481, 2020.

[37] Z. Lv, T. Liu, C. Shi, J. A. Benediktsson, and H. Du, "Novel land cover change detection method based on k-means clustering and adaptive majority voting using Bitemporal remote sensing images," *IEEE Access*, vol. 7, pp. 34425–34437, 2019.

[38] S. Di, Z.-L. Li, R. Tang, X. Pan, H. Liu, and Y. Niu, "Urban green space classification and water consumption analysis with remote-sensing technology: a case study in Beijing, China," *International Journal of Remote Sensing*, vol. 40, no. 5–6, pp. 1909–1929, 2019.