



Land Cover/Land Use Mapping of LISS IV Imagery Using Object-Based Convolutional Neural Network with Deep Features

S. Rajesh¹ · T. Gladima Nisia² · S. Arivazhagan³ · R. Abisekaraj⁴

Received: 10 October 2018 / Accepted: 24 October 2019 / Published online: 11 November 2019
© The Author(s) 2019

Abstract

The paper proposes a new method for classifying the LISS IV satellite images using deep learning method. Deep learning method is to automatically extract many features without any human intervention. The classification accuracy through deep learning is still improved by including object-based segmentation. The object-based deep feature learning method using CNN is used to accurately classify the remotely sensed images. The method is designed with the technique of extracting the deep features and using it for object-based classification. The proposed system extracts deep features using pre-defined filter values, thus increasing the overall performance of the process compared to randomly initialized filter values. The object-based classification method can preserve edge information in complex satellite images. To improve the classification accuracy and to reduce complexity, object-based deep learning technique is used. The proposed object-based deep learning approach is used to drastically increase the classification accuracy. Here, the remotely sensed images were used to classify the urban areas of Ahmadabad and Madurai cities. Experimental results show a better performance with the object-based classification.

Keywords Convolutional neural network (CNN) · Deep learning · LISS IV image · Image classification · Multi-resolution segmentation

Introduction

Classifying different areas of remote sensing image has a wide variety of applications in fields such as land cover mapping and detection, water resource detection, agricultural usage, wetland mapping, geological information and urban and regional planning. However, the classification of remote sensing images remains a tedious task due to its complexity. Feature extraction plays an important role in

classification. When features are chosen manually with human intervention, the efficiency of the classification process decreases. So, in order to improve the efficiency, we adopt an automatic feature learning method such as deep learning.

Deep learning is one of the excellent methods in artificial intelligence to learn discriminative features without human intervention. The work by LeCun et al. (2015) shows how deep learning is applied for classification. Unlike low-level feature representations, deeply learned features are generally more robust, and it has great effectiveness in image classification, such as face recognition (Sun et al. 2014) and scene classification (Li et al. 2010; Zhang et al. 2016a, b). In the remote sensing field, several researches are done for image classification using deep learning models, such as stacked autoencoder (SAE) and convolutional neural network (CNN). But, the original SAE extracts only one-dimensional spectral features which probably is not sufficient for high-resolution image classification. Therefore, another work is proposed (Chen et al. 2014b), which improved the SAE model by introducing

✉ S. Rajesh
srajeshfrank@yahoo.com

¹ Department of Information Technology, Mepco Schlenk Engineering College, Sivakasi 626005, India

² Department of Computer Science and Engineering, AAA College of Engineering and Technology, Sivakasi 626005, India

³ Department of ECE, Mepco Schlenk Engineering College, Sivakasi 626005, India

⁴ Naval Science and Technological Laboratory, DRDO, Vishakapatnam 530 627, India

spatial features for the efficient classification of hyperspectral images.

The CNN algorithm is popular for high-resolution image classification due to its effectiveness in spatial feature exploration (Zhao and Du 2016a, b; Zhao et al. 2015; Yue et al. 2015; Chen et al. 2014a). However, deep features are usually learned from local images patches and are pixel based, which sometimes may lead to misinterpretations. Also, deep feature extraction does not provide clear information about boundary and edges. So, to overcome the difficulty, the object-based classification is used instead of pixel-based classification. The object-based approach can be used to efficiently delineate and classify high-resolution imagery (Duveiller et al. 2008). Object-based classification method considers each image segments as building blocks for the further image analysis..

Now, let us look about the earlier works carried out for the classification of LISS IV images. The wavelet packet transform for texture analysis is presented for LISS IV image (Rajesh et al. 2011). Here, the statistical and co-occurrence features of the input patterns are first extracted and those features are used for classification. The Daubechies (DB2) wavelet filter is used for decomposition, and Mahalanobis distance classifier is used as the classifier. Since the methodology results in many features, some of which are found to be not useful, the best among the wavelet packet statistical and wavelet packet co-occurrence textural feature sets is selected using genetic algorithm (Rajesh et al. 2013). It outperforms the other feature reduction techniques like principal component analysis and linear discriminant analysis. Here, multilayer perceptron (MLP) layer is used for classification. The classification outperforms when the MLP for classification is replaced by adaptive neuro-fuzzy inference system (ANFIS) trained with backpropagation gradient descent method in combination with the least squares method (Rajesh and Arivazhagan 2014).

Our proposed system replaces the so-far carried-out works for classification of LISS IV image with a deep learning approach. CNN is used as a feature extractor, and deep features are obtained from LISS IV image. In order to still increase the efficiency, the obtained deep features are combined with the object-based textural features.

The organization of this paper is as follows: “[Study Area and Data Used](#)” section presents the study data and area. “[Convolutional Neural Network \(CNN\)](#)” section gives a simple introduction to convolutional neural network (CNN). “[Proposed Method](#)” section describes briefly about the proposed work and implementation. “[Experimental Study and Results](#)” section discusses the experimental study and results obtained. The conclusions are presented in “[Conclusion](#)” section.

Study Area and Data Used

For a supervised method of classification, the ground truth data are very important and it should be part of the training dataset, on the basis of which the real classification can be performed and tested. From the ground truth image of Ahmadabad shown in Fig. 1, the regions of bench mark data shown in Fig. 2 are labeled as 4 classes—, urban area, water and land area. The vegetation class includes the heavy and sparse vegetation areas in the image. The urban area class includes buildings, concrete and roads. The water class includes water in the given image. Land area class includes any area like barren farms, land areas between buildings, etc.

The multi-temporal satellite sensor images used in this study are of Madurai city in Tamil Nadu, state of India, and are shown in Fig. 3. Madurai is identified as one of the 12 heritage cities of India. The land cover features of this study area include urban, vegetation, water body, waste land and hilly region. The scene details of the area are as follows:

Satellite/sensor:	IRS P6/LISS IV
Resolution:	5.8 m
Band 2 (green):	0.52–0.59 μm
Band 3 (red):	0.62–0.68 μm
Band 4 (near-infrared):	0.77–0.86 μm

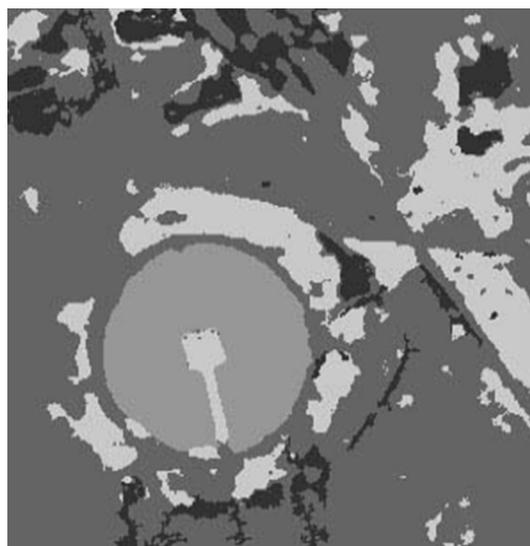


Fig. 1 Ground truth of Ahmadabad city



Fig. 2 Ahmadabad LISS IV image (300 × 300)

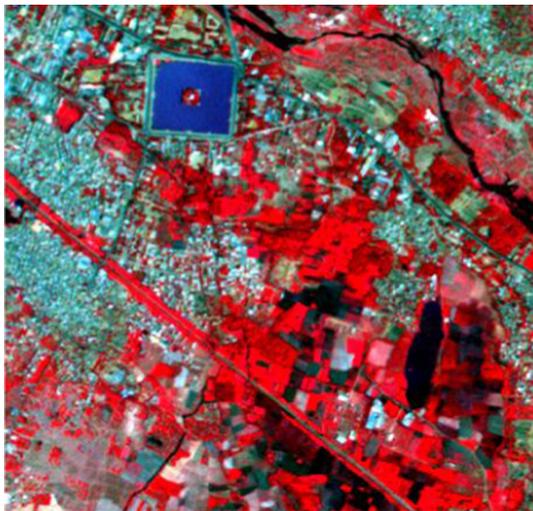


Fig. 3 Madurai city (size 400 × 400)

Convolutional Neural Network (CNN)

CNN is a deep, feed-forward neural network (Goodfellow et al. 2016). It consists of input and output layers; between them there are several multiple hidden layers. It is used primarily to classify images, cluster them by similarity and perform object recognition within scenes. They are algorithms that can identify many aspects of visual data. The architecture of the CNN is shown in Fig. 4.

In our proposed system, the CNN is used to extract high-level features automatically from the image with the help of manually initialized filters. The building blocks for CNN include convolutional layer, pooling layer, ReLU layer and fully connected layer. Usually, the CNN randomly initializes the filters and performs convolution operation. But the

proposed system uses filters such as mean filter, Gaussian filter, Sobel filter, disk filter, log filter, Prewitt filter, Laplacian filter as initial filters. The CNN has two important concepts: Locally connected network and parameter sharing. If we use the fully connected network, then there will be large number of parameters required. So locally connected networks are used through which parameter requirement is reduced. Also, by using parameter sharing, instead of using new parameters every time the already used parameters are utilized.

Proposed Method

The proposed method is introduced to classify objects in high-resolution images using object-based deep learning classification. The flowchart for object-based image classification combined with deep features is shown in Fig. 5. The proposed method is explained in the following sections.

Multi-resolution Segmentation

Object-based classification method improves the efficiency of classifying remotely sensed images. Segmentation groups the similar regions, and also it preserves the boundary information. The proposed system utilizes multi-resolution segmentation algorithm (Li et al. 2014) to segment image object. Scale, shape and compactness are important in segmenting image object. Scale, shape and compactness determine the size, shape and color of the image object, respectively. Objects of interest appear on different scales in an image. Therefore, the scale of resulting image objects should be freely adaptable to fit to the scale of task. Here, each pixel is considered as one image object. The image objects are merged based on local homogeneity, which describes the similarity of adjacent image objects, and larger image objects are obtained. The main components of multi-resolution segmentation are decision heuristics and definition of homogeneity of image objects. The decision heuristics determines the objects going to be merged at each stage, and definition of homogeneity of image objects is to compute the degree of fitting for a pair of image objects.

Extraction of Texture Feature

In texture training, the known texture features are mean, standard deviation, root mean square, energy, cluster shade, cluster prominence, correlation. Those features are calculated from the co-occurrence matrix $C(i,j)$ using Eqs. (1)–(9). The spectral feature NDVI is calculated using Eq. (10). The proposed system aims at obtaining the

Fig. 4 Architecture of convolutional neural network

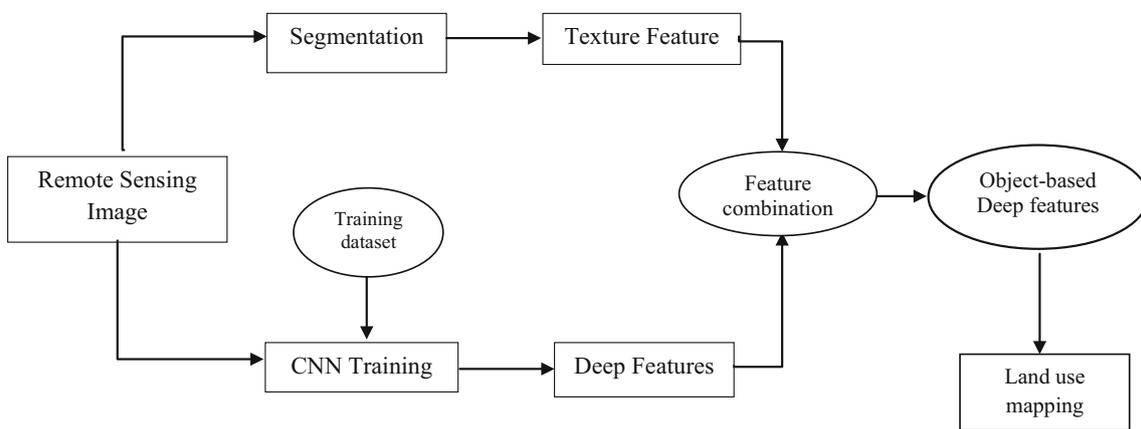
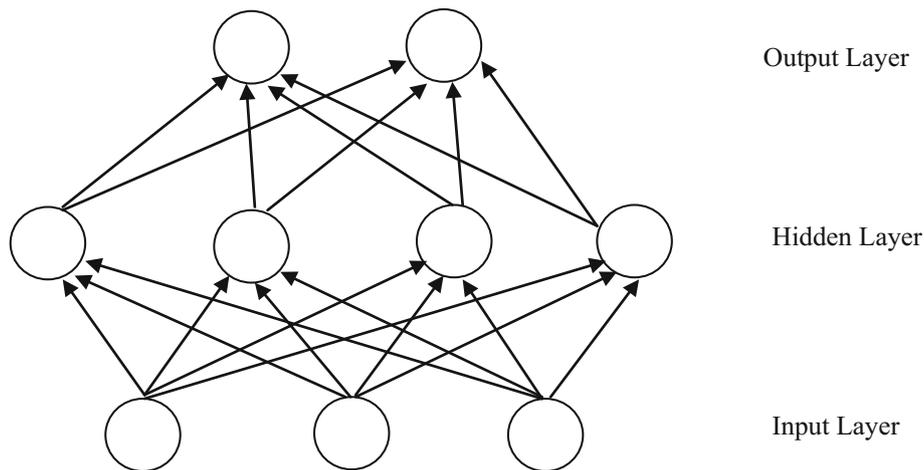


Fig. 5 Flowchart for object-based image classification combined with deep features

object-based texture features. So, the object-based segmented image is given as input and features are obtained using Eqs. (1)–(10):

$$\text{Mean } \bar{x} = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N x_{i,j} \tag{1}$$

$$\text{Variance } V = \frac{1}{N^2} \sum_{i=0}^N \sum_{j=1}^N (x_{i,j} - \bar{x})^2 \tag{2}$$

$$\text{Entropy} = - \sum_{i=1}^N \sum_{j=1}^N (C(i,j)) \log(C(i,j)). \tag{3}$$

$$\text{Contrast} = \sum_{i,j=0}^N (i-j)^2 C(i,j) \tag{4}$$

$$\text{Energy} = \sum_{i=1}^N \sum_{i=1}^N C(i-j)^2 \tag{5}$$

$$\text{Local homogeneity} = \sum_{i,j=0}^n 1/(1 + (i-j)^2) C(i,j) \tag{6}$$

$$\text{Cluster shade} = \sum_{i,j=0}^n (i - M_x + j - M_y)^3 C(i,j) \tag{7}$$

$$\text{Cluster prominence} = \sum_{i,j=0}^n (i - M_x + j - M_y)^4 C(i,j) \tag{8}$$

where $M_x = \sum_{i,j=0}^n iC(i,j)$ and $M_y = \sum_{i,j=0}^n jC(i,j)$

$$\text{Correlation} = \frac{\sum_{i=1}^N \sum_{j=1}^N [ijC(i,j)] - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{9}$$

where

$$\mu_x = \sum_i^N i \sum_j^N C(i,j)$$

$$\mu_y = \sum_j^N j \sum_i^N C(i,j)$$

$$\sigma_x^2 = \sum_i^N (a - \mu_x)^2 \sum_j^N C(i,j)$$

$$\sigma_y^2 = \sum_j^N (b - \mu_y)^2 \sum_i^N C(i,j)$$

$$NDVI = \frac{(\text{near IR band} - \text{red band})}{(\text{near IR band} + \text{red band})} \tag{10}$$

These features are extracted, and thus, the feature set is formed. The obtained feature set represents the object-based texture feature since the segmented image is given as input for feature extraction.

Deep Feature Extraction

Deep features consist of *n*-number of convolution and pooling. The illustration of convolutional neural network-based framework is shown in Fig. 6. The main reason for using convolution in the proposed system is to extract deep features from the input LISS IV image. The convolutional layer applies the convolution operator to the input image and passes its output to the next layer. Here, a filter is made to slide or convolve around the input image. Every unique location on the input produces a value. The weight matrix helps in extracting important information from image matrix. The output matrix obtained is called the ‘convolved feature’ or ‘activation map.’

$$f(\mathbf{X}; \theta) = W \cdot H \tag{11}$$

where *W* represents filters, *X* represents training samples, the filters and biases are collectively referred as θ , and *H* represents hidden layers. The filters are used to extract

features. The size of the activation map is controlled by two parameters; they are depth and stride.

Depth It refers to the number of filters used for the convolution operation.

Stride It refers to the number of pixels sliding the weight matrix over the input matrix.

Pooling is to reduce the dimensionality of the activation map. It can be of different types, namely max, average, sum:

$$h = \text{pool}(g(W \cdot H)). \tag{12}$$

The final pooling, *h*, represents the deep feature, and *g* represents a point-wise nonlinear tanh activation function. Thus, the final deep features are obtained.

The extracted texture features (from Eqs. 1 to 10) such as mean, standard deviation, root mean square, energy, cluster shade, cluster prominence, correlation and deep features are combined and used for classification.

Object-Based Classification with Deep Features

An image *I* contains *N* image objects $O_i, i \in \{1, 2, \dots, N\}$, and there are *M* pixels $I_j, j \in \{1, 2, \dots, M\}$ inside object O_i . For each pixel I_j , the deep features are denoted as F_j . Similarly, the object-based texture features (such as NDVI) are represented as R_j . For feature combination purpose, the deep features are combined with object-based properties $U_j = [F_j, R_j]$ for joint feature classification of the original imagery at the pixel level. Then, the label of an object O_i is predicted from the majority statistics of feature vectors U_j using gaussian function in CNN (Zhao et al. 2017). The obtained U_j is used for calculating the predicted label:

$$\text{Predicted label} = \exp\left(-\frac{1}{2} (U_j^i)^2\right).$$

The CNN training is carried out with one or multiple layers. The training is carried out by applying methods such as backpropagation methods and steepest gradient descent method. Algorithm for backpropagation is as follows:

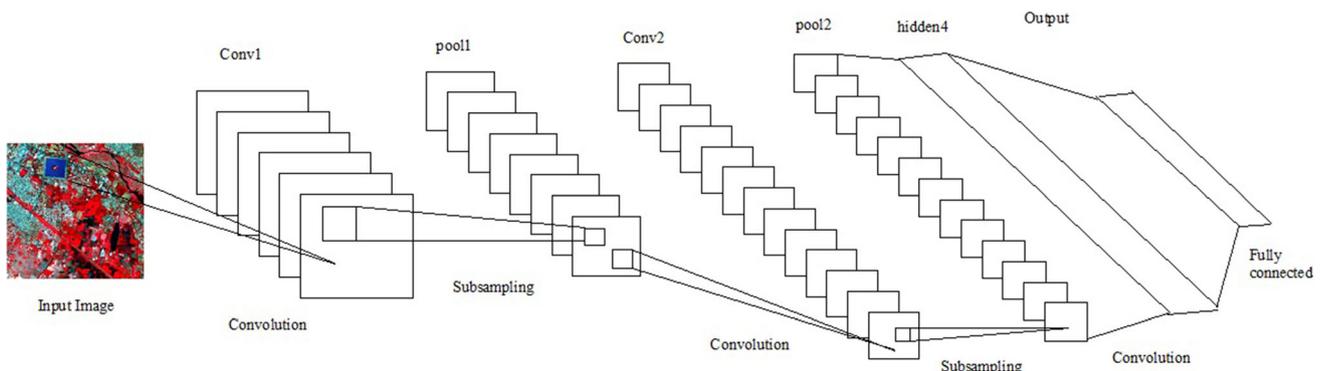


Fig. 6 Illustration of convolutional neural network

Algorithm for Training CNN:

```

// Input : LISS IV Remotely sensed training image set, Ti and expected probability
//Output : Trained CNN

Initialize filter weight, number of epoch
Design a CNN
For each image in Ti
    Initialize i=1
    Repeat until obtained probability  $\approx$  expected probability
        Train CNN using Ti
        C = Convolution of Ti
        P = Perform pooling for C
        R = Apply ReLu to P
        Send R features to fully connected layer
        Obtain probability value for each class using softmax layer
    end repeat
    Final label for each object is obtained
end for
return trained CNN

```

The trained CNN is then used to classify the test image. Thus, the different portions of the remotely sensed image are classified and labeled efficiently.

Experimental Study and Results

The proposed method is implemented using Matlab R2018b. The section presents the performance analysis of the proposed method to classify the remotely sensed LISS IV images. Two satellite images captured by Indian Remote Sensing satellite IRS-P6 are taken for analysis. These images are subjected to geometric correction with the help of ground control points (GCPS). As mentioned above, we combine deep feature and object-based classification method for accurate high-resolution image interpretation.

Performance of Proposed Method to Ahmadabad LISS IV Image

The satellite imagery used corresponds to areas in and around Ahmadabad city. The image is a part of IRS P6 LISS IV imagery which has a spatial resolution of 5.8

meters. For Ahmadabad city, we have 4 different classes such as urban area, Vegetation, Water body and Land cover. The size of Ahmadabad LISS IV image is 300×300 pixels. For the input training samples, the size of training sample is 7×7 pixel. The testing is carried for different layers of configuration CNN. The one-layer CNN only has a single convolution layer. Similarly, the two-layer CNN, three-layer CNN and four-layer CNN have two convolution layers, three convolution layers and four convolution layers, respectively. Thus, the classification accuracy for different CNNs is calculated as shown in Table 1. The performance of different configuration CNNs is shown in Table 2. The Ahmadabad dataset is a multi-spectral image, where the first convolutional layer has 20 filters of 3×3 ($3 \times 3 \times 3$). The output of convolutional

Table 1 Classification accuracy of Ahmadabad dataset

CNNs	WB	VEG	UB	LC
One-layer	89	70	84	78
Two-layer	89	78	88	82
Three-layer	91	84	88	86
Four-layer	92	90	92	90

Table 2 Performance of different configuration CNNs—Ahmadabad dataset

CNNs	Precision (%)	Recall (%)	Accuracy (%)	F1 (%)
One-layer	80	82	80.5	81
Two-layer	86	86	84.5	86
Three-layer	88	87	87	87
Four-layer	90	90	90.5	90

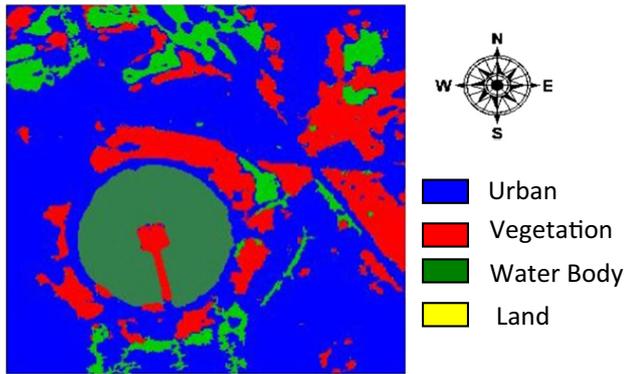


Fig. 7 Classified output of Ahmadabad using CNN

layer is passed through max pooling (2 * 2), and it is moved through second convolutional layer with 50 filters (3 * 3 * 20) and in third convolutional layer with 70 filters (3 * 3 * 40) and in fourth layer with 100 filters (3 * 3 * 60). The input image of size 7 * 7 pixel, after the first convolution the output size of the image is 10 * 10 pixel with 20 filters and after the second convolution, the output size of the image is 13 * 13 pixel with 50 filters and after the third convolution, the output size of the image is 16 * 16 pixel with 70 filters and after the fourth convolution, the output size of the image is 19 * 19 pixel with 100 filters.

To demonstrate the effectiveness of the proposed technique, a total of 3550 sample image regions were taken for

training. The sampling strategy followed is random sampling. The entire image of Fig. 2 is given for testing, and the output is shown as multispectral image in Fig. 7. The performance analysis of these techniques has been carried out in terms of overall, producer, user and Kappa coefficient accuracy indices. The accuracy indices are carried for all the labels of Ahmadabad city. Producer accuracy, user accuracy and Kappa coefficient are calculated for the classified output image. Producer accuracy is the ratio between number of correctly classified pixels in each category to the number of training pixels used for the category. Water body has the highest producer accuracy of 98%, and vegetation, urban area and land cover have 97%, 96% and 96%, respectively. User accuracy is the ratio between number correctly classified pixels to the total number of pixels belonging to that particular class in the confusion matrix. Water body, land cover, vegetation and urban area have the user accuracy of 98%, 96%, 95% and 95%, respectively. Kappa coefficient has been calculated, and it results with water body (0.85), vegetation (0.62), urban area (0.89) and land cover (0.75). The overall accuracy for Ahmadabad city is 90.25%. Table 3 shows the confusion matrix results obtained by comparing the reference map with the classified map, for the proposed system.

Performance of Proposed Method to Madurai LISS IV Image

Madurai dataset has different classes such as waste land, water body, urban area, vegetation and saline land. The size of Madurai LISS IV image is 400X400 pixels. For the input training samples, the size of training sample is 7 * 7 pixel. Here, 300 samples are taken for training and 500 samples are taken for testing. The testing is carried for different layers of configuration CNN. The classification accuracy is calculated and is shown in Table 4. The performance of different configuration CNNs is shown in

Table 3 Confusion matrix for Ahmadabad city image

Classified Data	Number of Samples	Reference Data			
		Veg	Water body	Urban	Land
Vegetation	9628	9146	80	317	85
Water body	16563	92	16232	119	120
Urban	53228	773	103	50567	1785
Land Cover	7017	107	19	155	6736
User Accuracy : 96		Producer Accuracy : 96.75			
Overall Accuracy : 90.25		Kappa Coefficient : 0.78			

Table 4 Classification accuracy of Madurai dataset

CNNs	UB	VEG	WB	WL	SL
One-layer	91.1	88.8	100	80	71.3
Two-layer	93	93	100	80	77
Three-layer	95	93	100	86	86
Four-layer	95	93	100	91	84

Table 5 Performance of different configuration CNNs—Madurai dataset

CNNs	Precision (%)	Recall (%)	Accuracy (%)	F1 (%)
One-layer	86	87	86	87
Two-layer	90	87	88	89
Three-layer	93	91	89	92
Four-layer	94	92	92	93

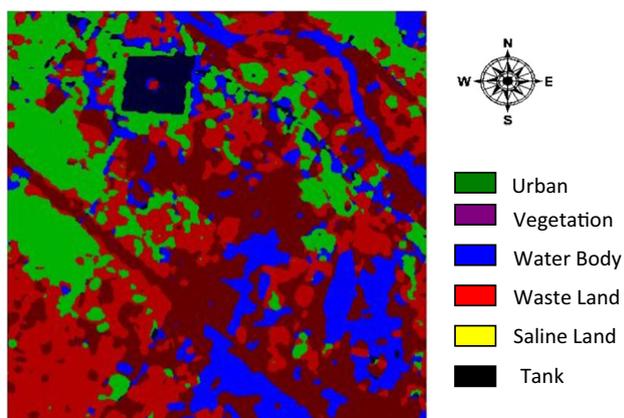


Fig. 8 Classified output of Madurai using CNN

Table 5. The Madurai dataset is a colored image, where the first convolutional layer has 20 filters of $3 * 3 (3 * 3 * 3)$. The output of convolutional layer is passed through max pooling ($2 * 2$), and it is moved through second convolutional layer with 50 filters ($3 * 3 * 20$) and in third convolutional layer with 70 filters ($3 * 3 * 40$) and in fourth layer with 100 filters ($3 * 3 * 60$). The input image of size $7 * 7$ pixel, after the first convolution the output size of the image is $10 * 10$ pixel with 20 filters and after the second convolution, the output size of the image is $13 * 13$ pixel with 50 filters and after the third convolution, the output size of the image is $16 * 16$ pixel with 70 filters and after the fourth convolution, the output size of the image is $19 * 19$ pixel with 100 filters.

The classified output of Madurai LISS IV image is shown in Fig. 8. The experimental study was carried for LISS IV Madurai image. The sampling strategy followed here is random sampling. The number of training samples was chosen based on the heuristic $N_i * x (N_i + 1)$ given by Kavzoglu and Mather, where N_i is the number of neurons. To test the efficiency, 500 samples were selected randomly from the study area. The 500 samples (pixels) chosen are 131, 216, 66, 59 and 4 pixels of urban, vegetation, water body, waste land and saline land samples, respectively. About 500 Ground Control Points (GCPs) were used to create the reference dataset for the assessment using *e-Trex* venture global positioning system device. Some of the GCPs collected during field survey of Madurai city. Few of the GCPs are listed out in Table 6.

The performance analysis of these techniques has been carried out in terms of overall, producer, user and Kappa coefficient accuracy indices. The accuracy indices are carried for all the labels of Madurai city. Producer accuracy is calculated for LISS IV Madurai image. Water body and

Table 6 Ground control points of Madurai city

Name of the area	Latitude	Longitude	Elevation (Feet)	Accuracy (Feet)	Features
Kudhal nagar tank	9°57'03.9"	78°06'14.8"	138	24	Vegetation
Sellur tank	9°56'26.4"	78°07'06.3"	148	27	Water body
Malar nagar	9°57'25.2"	78°06'52.3"	140	25	Waste land
Alangulam tank	9°57'25.6"	78°07'14.4"	138	24	Water body
Reserve line	9°56'49.8"	78°07'56.1"	140	26	Urban
Burma colony	9°51'20.5"	78°06'07.1"	148	21	Urban
Perungudi	9°50'33.9"	78°05'33.2"	134	24	Waste land
Ring road 1	9°51'23.4"	78°07'08.5"	127	22	Waste land
Ring road 2	9°51'31.7"	78°07'15.0"	125	22	Waste land
Chinthamani	9°53'14.8"	78°08'37.8"	133	26	Urban
Melamadai	9°55'38.9"	78°09'06.7"	135	25	Waste land
Anna nagar	9°55'04.4"	78°08'55.3"	140	29	Urban
Vandiyur stop	9°54'35.3"	78°09'36.5"	122	26	Urban
Vandiyur extend	9°54'24.0"	78°09'47.3"	126	24	Waste land
Vaigai river	9°54'45.3"	78°09'20.1"	122	23	Water body

Table 7 Confusion matrix for Madurai image

Classified Data	Number of Samples	Reference Data				
		Urban	Veg.	Water body	Waste land	Saline Land
Urban	131	127	1	0	3	0
Vegetation	216	4	207	0	3	2
Water body	66	1	0	64	0	1
Waste Land	59	2	1	1	55	0
Saline Land	4	0	0	0	1	3
User Accuracy : 94.6		Producer Accuracy : 96.4				
Overall Accuracy : 92		Kappa Coefficient : 0.85				

Urban area have the producer accuracy of 97%. Vegetation, waste land and saline land have the accuracy of 96%. User accuracy is also calculated. Water body and urban area have the accuracy of 97%. Vegetation, waste land and Saline land have the accuracy of 96%, 93% and 90%, respectively. Kappa coefficient calculated produced the results such as water body (0.95), waste land (0.89), urban area (0.83), vegetation (0.75) and saline land (0.8). The overall accuracy for Madurai city is 92%. Table 7 shows the confusion matrix results obtained by comparing the

reference map with the classified map, for the proposed system.

To compare the accuracy of different algorithms, classified output of Madurai LISS IV image with DB2, DB2 with GA and ANN, DB2 with GA and ANFIS and object-based deep learning methods is shown in Fig. 9. When comparing all the algorithms, object-based deep learning technique has classified the image more accurately. From Table 7, which shows the classification results of different

Fig. 9 **a** Object-based deep learning technique, **b** DB2, **c** DB2 with GA and ANN, **d** DB2 with GA and ANFIS

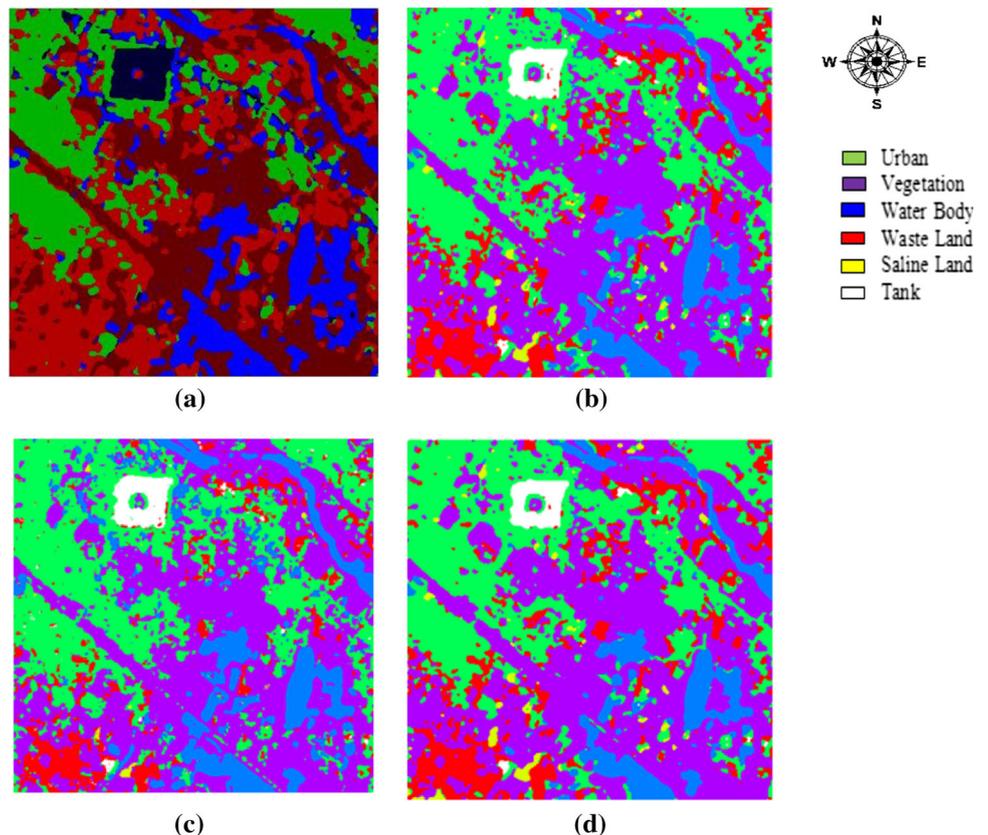


Table 8 Classification results of different algorithms

	Overall	Producer	User	Kappa
DB2 with ANN	83.62	84.00	83.10	0.80
DB2 with ANFIS	86.01	85.19	84.28	0.80
DB2	87.60	89.57	82.02	0.82
Object-based deep learning	92	96.4	94.6	0.85

algorithms, it is evident that object-based deep learning technique produces the best results.

Conclusion

The system utilized the CNN for the effective classification of high-resolution LISS IV images using object-based strategy. Feature extraction is one of the toughest tasks in the analysis of the images. The challenge was efficiently handled by choosing CNN for automatic feature learning with the help of pre-defined filter values. We propose an effective way to classify high-resolution LISS IV images by combining deep features with image objects. The deep features are evaluated by testing the CNN framework with five different layer configurations for the classification of images. The proposed approach significantly reduces the complexity of the feature set and also improves the classifier performance when compared to earlier approaches. The results obtained by the proposed method on Ahmadabad City remotely sensed image and Madurai city remotely sensed image demonstrate that the proposed method outperforms other studied methods.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Chen, X., Xiang, S., Liu, C. L., & Pan, C. H. (2014a). Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 11(10), 1797–1801.
- Chen, Y., Lin, Z., Zhao, X., Wang, G., & Gu, Y. (2014b). Deep learning-based classification of hyperspectral data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6), 2094–2107.
- Duveiller, G., Defourny, P., Desclée, B., & Mayaux, P. (2008). Deforestation in Central Africa: Estimates at regional, national and landscape levels by advanced processing of systematically-

- distributed landsat extracts. *Remote Sensing of Environment*, 112(5), 1969–1981.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge: MIT Press.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- Li, H., Tang, Y., Liu, Q., Ding, H., Jing, L., Lin, Q. (2014). A novel multi-resolution segmentation algorithm for high resolution remote sensing imagery based on minimum spanning tree and minimum heterogeneity criterion. In *Proceedings of IEEE international geoscience remote sensing symposium* (vol. 3, pp. 2850–2854).
- Li, L.-J., Su, H., Fei-Fei, L., & Xing, E. P. (2010). Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *Proceedings of the neural information processing systems conference* (pp. 1378–1386).
- Rajesh, S., & Arivazhagan, S. (2014). adaptive neuro-fuzzy inference system based land cover/land use mapping of LISS IV imagery using wavelet packet transform. *Journal of the Indian Society of Remote Sensing*, 42(2), 267–277.
- Rajesh, S., Arivazhagan, S., Pradeep Moses, K., & Abisekaraj, R. (2011). Land cover/land use mapping using different wavelet packet transforms for LISS IV Madurai imagery. *Journal of the Indian Society of Remote Sensing*, 40(2), 313–324.
- Rajesh, S., Arivazhagan, S., Pradeep Moses, K., & Abisekaraj, R. (2013). Genetic algorithm based feature subset selection for land cover/land use mapping using wavelet packet transform. *Journal of the Indian Society of Remote Sensing*, 41(2), 237–248.
- Sun, Y., Chen, Y., Wang, X., & Tang, X. (2014). Deep learning face representation by joint identification- verification. In *Proceedings of advances neural information processing system conference* (pp. 1988–1996).
- Yue, J., Zhao, W., Mao, S., & Liu, H. (2015). Spectral-spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sensing Letters*, 6(6), 468–477.
- Zhang, F., Du, B., & Zhang, L. (2016a). Scene classification via a gradient boosting random convolutional network framework. *IEEE Transaction on Geoscience and Remote Sensing*, 54(3), 1793–1802.
- Zhang, F., Du, B., Zhang, L., & Xu, M. (2016b). Weakly supervised learning based on coupled convolutional neural networks for aircraft detection. *IEEE Transaction Geoscience and Remote Sensing*, 54(9), 5553–5563.
- Zhao, W., & Du, S. (2016a). Spectral-spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach. *IEEE Transaction on Geoscience and Remote Sensing*, 54(8), 4544–4554.
- Zhao, W., & Du, S. (2016b). Learning multiscale and deep representations for classifying remotely sensed imagery. *ISPRS Journal Photogrammetry Remote Sensing*, 113, 155–165.
- Zhao, W., Guo, Z., Yue, J., Zhang, X., & Luo, L. (2015). On combining multiscale deep learning features for the classification of hyperspectral remote sensing imagery. *International Journal of Remote Sensing*, 36(13), 3368–3379. <https://doi.org/10.1080/2150704X.2015.1062157>.
- Zhao, W., Shihong, D., & Emery, W. J. (2017). Object-based convolutional neural network for high-resolution imagery classification. *IEEE Journal of Selected topics in Applied Earth observation and Remote Sensing*, 10(7), 3386–3396.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.