International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 3 | ISSN : 2456-3307

Deep Convolutional Neural Network Models for Land Use and Land Cover Identification Using Dataset Created From LISS-IV Satellite Images

Parminder Kaur Birdi[*1], Karbhari Kale[2]
[1]MGM's Jawaharlal Nehru Engineering College, N-6, CIDCO, Aurangabad, Maharashtra, India
[2]Department of CS & IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, Maharashtra, India

ABSTRACT

Identification of land covers like crop-land, settlement, water-body and others from remote sensing images are useful for applications in the area of rural development, urban sprawl etc. In this paper we are addressing the task of identification of different land covers using remote sensed images which is further useful for image classification. Deep learning methods using Convolutional Neural Networks (CNN) for remote sensed or satellite image classification is gaining a strong foothold due to promising results. The most important characteristic of CNN-based methods is that prior feature extraction is not required which leads to good generalization capabilities. In this paper firstly we are presenting dataset prepared using multispectral, high-resolution images from LISS-IV sensor and another dataset of PAN images created using coarse-resolution images from Landsat-8 sensor. LISS-IV dataset is prepared for six commonly found land covers i.e. crop-land, water-body, bare-farm, road and settlement. Secondly we are proposing two patch-based Deep Convolutional Neural Networks (DCNN) models for prediction/identification of the land covers present in the image. Experiments conducted using the LISS-IV dataset has shown promising accuracies on both the DCNN models. Implementation of network is made efficient by harnessing graphics processing unit (GPU) power which reduces computation time. And finally, DCNN models are also evaluated for their performance using two similar publicly available benchmarked datasets, indicating that construction of models using described size of filters, number of filters and number of layers is suitable for multi-class remote sensing image patch prediction or identification.

Keywords: LISS-IV dataset, Patch-based learning, Convolutional neural networks, Test accuracy, land-use, land covers.

I. INTRODUCTION

A large number of satellites capturing huge amount of images can be used for wide range of applications for land-use analysis, agriculture planning and rural & urban development. Visual world can be recognized better by using proper representation of objects. From an image, primitive features can be extracted followed by different parts of the object which leads to identification of the object. This is the key motivation behind deep learning, a branch of Machine Learning where neural networks are constructed with more than one hidden layer called as Deep learning Networks. For image classification, deep learning methods using Convolutional Neural Networks (CNN) can be applied to develop generalized algorithms which can be used for solving problems of different domains. The most important characteristic of CNN-based methods is that prior feature extraction is not required which leads to good

generalization capabilities. CNNs have shown good performance in object recognition, object detection and remote image classification [1, 2, 3].

CNNs are inspired by the working of visual system of human beings where we do visual perception of things present around us using a layered architecture of neurons [4]. Hand-crafted feature extraction was the main method used for image classification, as reflected in most of the traditional image classification. CNN can learn suitable internal representations of the images. By using CNNs, learning models are able to obtain conceptual sensitivities by each layer. CNNs work as feature extractors and classifiers which are trainable as compared to traditional classifiers which makes use of hand-crafted features.

The key contributions of this paper:

1. Creation of new dataset using satellite images from high resolution Linear Imaging Self-Scanner (LISS-IV), consists of 2500 image scenes which is increased to 18000 images using data augmentation, labelled for six different classes. This dataset can be used for training of CNNs for land use and land cover classification tasks. Another dataset is created using PAN band of Landsat8 Images. Ten images acquired over a period of one year have been used to create the dataset. This dataset is small with only 1000 image scenes and could be created only for three classes i.e. water, crop-area and settlement, due to small size of pixel. The PAN band of Landsat8 has spatial resolution of 15m.
2. Proposed a patch-based learning framework to design deep CNN (DCNN) models which can identify various land use and land covers present in a satellite image. Our study also shows that training process can be speed up by utilizing GPU power which will make it possible to scale up such models for larger inputs and train them on large datasets.

Further paper is arranged as follows. Section II is discussing related work carried out in the similar domain. Section III describes the methodology covering architecture of proposed DCNN models, study area and experimental setup. Section IV describes results and discussions followed by conclusions with future scope in section V.

II. RELATED WORK

Deep neural networks can learn effective feature representations from a big training dataset and these features can be used for classification purposes. This area of remote sensing area is progressing at slow pace because of less availability of labelled ground truth datasets. Yang and Newsam presented an intensively researched remote sensing image classification dataset known as UC Merced Land Use Dataset (UCM) [5, 22]. The dataset consists of twenty one land use and land cover classes. Each class has 100 images and the images are present in size of 256x256 pixels with a spatial resolution of about 30 cm per pixel. All images are in the RGB colour space and were extracted from the USGS National Map Urban Area Imagery collection. Helber et al., recently proposed a novel satellite image dataset for land use and land cover classification, EuroSAT [6]. This dataset consists of 27,000 labelled images and is created from Sentinel-2 satellite images. There are ten different classes present in the dataset namely, industrial, residential, annual crop, permanent crop, river, salt lake, highway, vegetation, pasture and forest. Image dataset is provided in two formats, one covering 3 bands and other one having 13 spectral bands.

A remote sensing image classification benchmark (RSI-CB) based on massive, scalable, and diverse crowd-source data has been presented [7]. Dataset has been labelled using Open Street Map (OSM) data, ground objects by points of interest and vector data from OSM leading to a large-scale benchmark for remote sensing image classification. This benchmark

has two sub-datasets with 256 × 256 and 128 × 128 sizes, as requirements of designed DCNNs, for image sizes. The first dataset has six categories with 35 subclasses of more than 24,000 images and the second one has also six categories with 45 subclasses of more than 36,000 images. The main categories are agricultural land, construction land, transportation, water, woodland, and other lands, along with their several subclasses. Others publicly available datasets are: WHU-RS19, having 1005 images for 19 categories with image patch size of 600x600, SIRI-WHU having 2400 images for 12 classes with image patch size of 200x200, RSSCN7 with 2800 images, patch size of 400x400 for 7 classes, RSC11 having 1232 images, patch size of 512x512 for 11 classes, Brazilian coffee scene with 2876 images, patch size of 64x64 for 2 classes i.e. coffee and non-coffee [8, 9, 10, 21, 22, 23, 24]. Interested authors can read comprehensive review of these datasets [11]. They have also proposed, NWPU-RESISC45, which can be used for remote sensing image classification. It has 31500 images, for 45 classes [11]. Images of this dataset has large number of image scenes with variations in translation, spatial resolution, viewpoint, object pose, illumination etc.

A convolutional neural network (CNN) has been applied to multispectral orthoimagery with spatial resolution of 0.5m and a digital surface model (DSM) of a small city producing fast and accurate per-pixel classification [12]. Authors evaluated and analysed various design choices of the CNN architecture. Finally it was concluded that CNNs are a feasible tool for solving both the segmentation and object recognition task. Two new satellite datasets called SAT-4 and SAT-6 have been prepared from images taken from the National Agriculture Imagery Program (NAIP) dataset [13]. Images are acquired in patch size of 28x28 and consists of four bands- red, green, blue and Near Infrared. They have also proposed a classification framework which extracts features from the normalized input image and these feature vectors are given as input to a Deep Belief Network for classification. Authors observed that for SAT-4 dataset, on their best network produced a classification accuracy of 97.95% and for SAT-6, it produced a classification accuracy of 93.9%. Yang and Newsam investigated bag-of-visual-words (BOVW) methods for land-use classification for high-resolution imagery [22]. Authors have proposed a novel method, termed as the spatial co-occurrence kernel which takes into account the relative arrangement. Methods are evaluated using a large ground truth image dataset, UC Mercedes consisting of 21 land-use classes. Authors concluded that even though BOVW methods do not always perform better than the standard approaches, but they can be used as a robust alternative which is more effective for certain land-use classes.

Supervised data mining methods like neural networks, support vector machines and random forests majorly use spectral information. This information change due to impact of weather conditions, sensor geometry etc. Information extraction process can be mechanized to overcome drawbacks of hand-crafted methods. High classification accuracies are obtained for very high spatial resolution images and mostly for the cases where training images are completely hand-labelled i.e. per-pixel. This kind of 100% labelled data is not available easily so we have taken the approach of patch-based dataset creation. In this approach, label is assigned to the entire patch as compared to labelling every pixel. We are addressing the challenge of less availability of labelled dataset to train a CNN by creating a new dataset built from multispectral sensor with spatial resolution of 5.8 m (which is much lower than the resolution of images used by majority of researchers) and also PAN band dataset from coarse-level resolution sensor, Landsat-8 having spatial resolution of 15m. We have also attempted to propose two new DCNN architectures with few number of convolutional layers which can be efficiently trained and tested using the new

proposed datasets and also compared their performance with standard benchmark dataset.

III. METHODOLOGY

A. Deep convolutional neural networks (DCNN) architecture

Deep convolutional neural networks have shown very good performance in experiments conducted on remote sensing images for classification or object detection purposes. Constructing a DCNN requires major steps: creating the convolutional neural network architecture, preparing training and test data and initializing parameters for training process. In CNNs/DCNNs, the properties related to the structure of layers, number of neurons, number & size of filter, receptive field (R), padding (P), the input volume dimensions (Width x Height x Depth, or N x N x B) and stride length (S) are called hyper-parameters [14, 15]. Connecting all the neurons with all possible areas of the input volume is a difficult task and leads to large number of weights to train. This results in a very high computational complexity. So, instead of connecting each neuron to all possible pixels, a 2- dimensional region, say of size 5×5 pixels is defined and it extends to the depth of the input, making receptive field size to be 5x5x3 (for 3 band input image). Computations are carried out for these receptive fields producing the activation map.

First step is to choose filter/ kernel of appropriate size to convolve over the input image. The main goal of this step is to identify key features in the image. This convolution operations produces activation maps. Activation maps represent 'activated' neurons/ regions, i.e. area where features specific to the kernel have been found in the input patch. Initialization of weight values to filter is done randomly here and then these values are updated with each learning iteration over the training set, as part of back propagation. Convolution operations find significant features like edges, lines and intensity, when appropriate filters are convolved over the image patch. Selection of proper size of the filters is very important step to identify the significant features. Therefore it's very important to find the appropriate size of the kernel/filter. In our design of DCNN, kernel size used are 5x5, 3x3 and 1x1 depending upon the input size of patch for that convolution layer. The keys points in designing a DCNN model are setting local connections and pooling. The main goal of pooling layer is to reduce the dimensionality of input data, also called as down-sampling. If pooling is removed, the dimensionality of the problem increases drastically leading to large training time. While deciding stride factor for pooling, care must be taken that it doesn't result in loss of information.

Any raw image of any size can be given as input to the algorithm designed which is first resized to the desired image patch size of 32x32. Resized image patch (multispectral image) is fed as input to DCNN. Image patch is represented as a 3D tensor of dimensions N x N x B, where N represents length & width of the image and B is number of bands/channels. Therefore, all the factors discussed above play a significant role in making deep networks get trained. Architectural building blocks of DCNN models are shown in Figure 1. In the models designed by us, for pooling layer, a stride factor of 2 is used. This layer is usually placed after convolution layer. Even though pooling results in some amount of information loss, it still is found beneficial for the network as reduction in size leads to less computational overhead for the upcoming layers of the network and it also work against over-fitting.

An important role in the training process is the choice of activation function, the way weights are initialized, and how learning is implemented. Activation functions are identity or linear function, sigmoid or logistic function, hyperbolic tangent and Rectified Liner Unit (ReLU). Major role is played by the choice of activation function, most widely used is

ReLU [16]. The output layer is the softmax layer which produces a set of output activations representing predicted probabilities which always sum to 1.

The function of the Softmax layer is to convert any vector of real numbers into a vector of probabilities, thus corresponding to the likelihoods that an input image is a member of a particular class. Batch normalization potentially helps in two ways: faster learning and higher overall accuracy. It is performed on the mini-batch size specified in the parameters. For normalization purposes, we divide the calculated value of the activation matrix by the sum of values in the filter matrix. Since there is a very large number of patches in our dataset we use stochastic gradient descent with mini batches for optimizing learning.

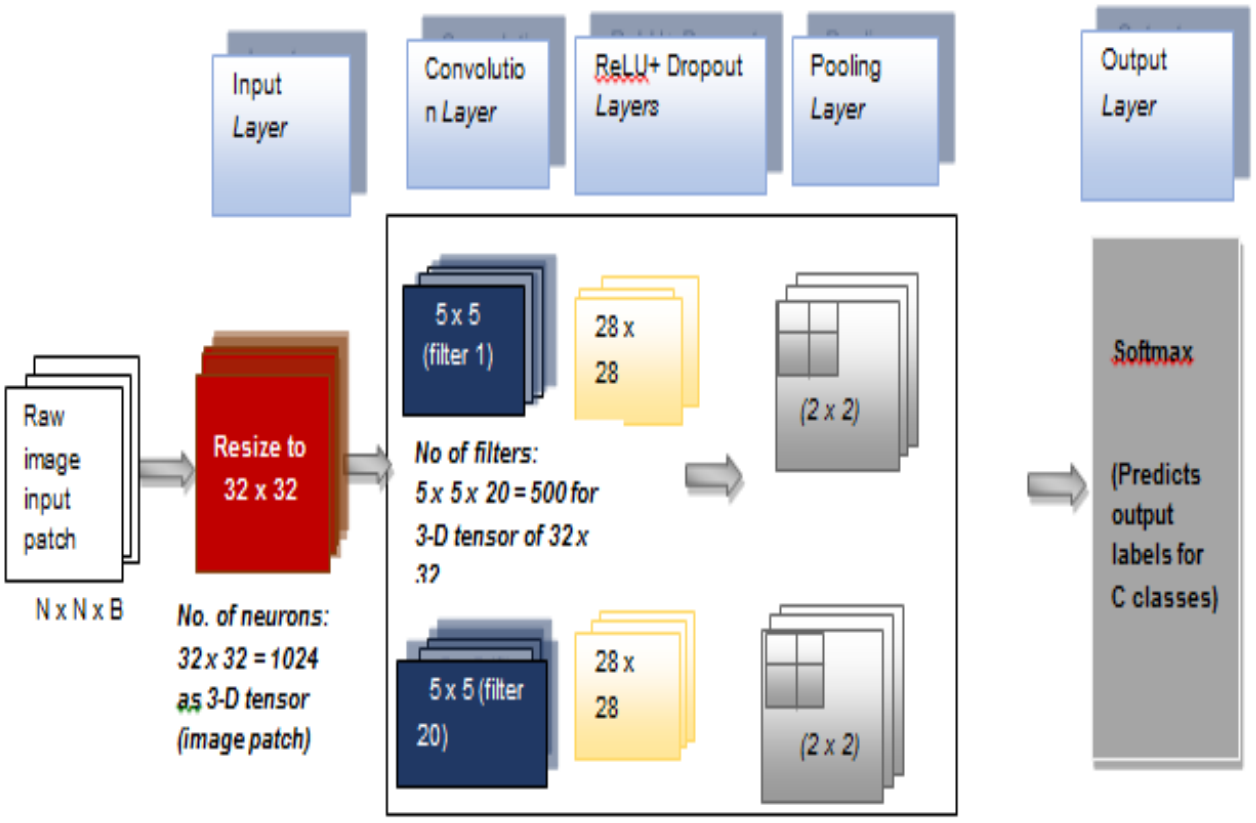

Figure 1. Architectural building blocks of DCNN models

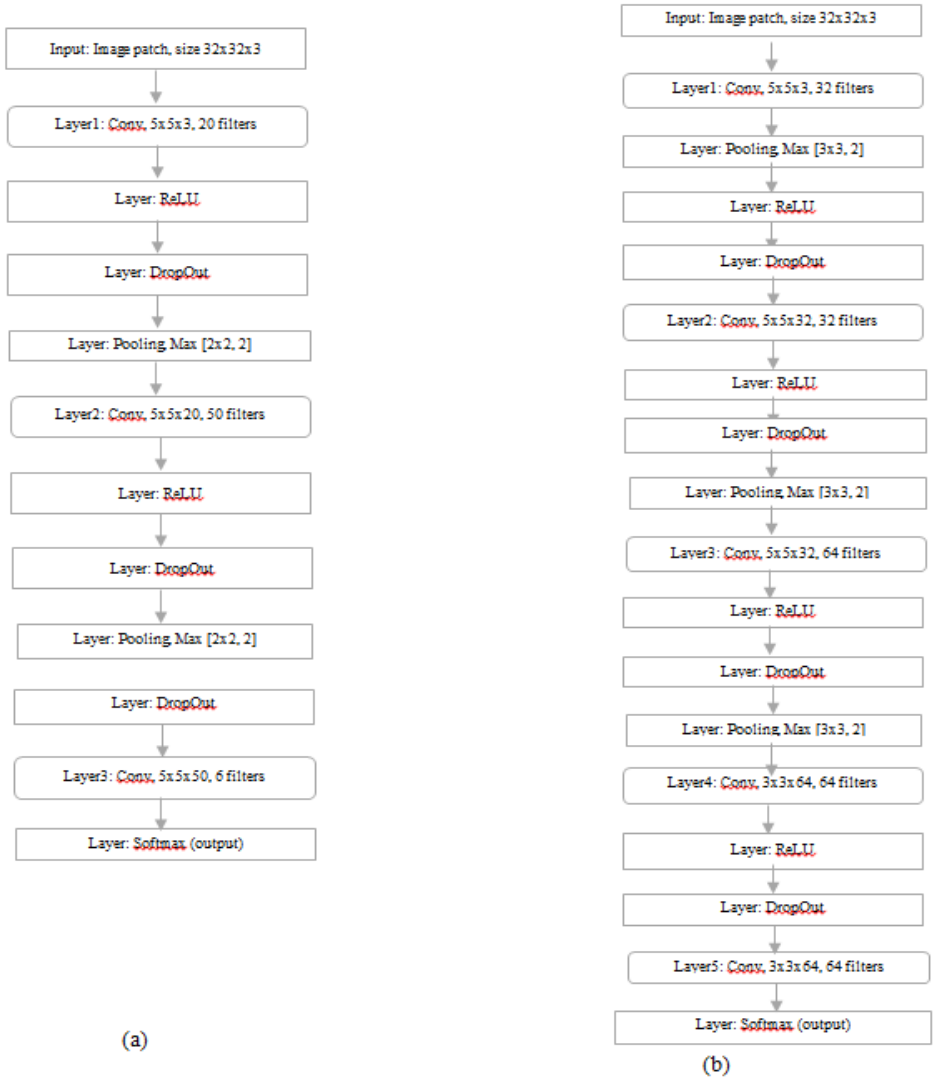

Figure 2. Deep Convolutional Neural Network Model 1(a) and Model 2 (b)

B. Creation of training and test labelled dataset

CNNs require large number of images for training for learning task. The image resolutions, size and scale of objects impacts the training process, as task-relevant information varies with spatial resolution. How an object is recorded in image depends on object's location, angle of capture and its size. This is to be considered for data augmentation. CNNs can deal with change in location as weights are shared in convolutional layers. Majority of the researchers have used hand-labelled dataset created for both training and testing and since labelling images is a very time consuming process, the datasets have been small in both aerial image applications and general image labelling work [12, 17, 18]. For this paper, the study site is Navin Kaigaon village of Maharashtra, India. The study area as shown in figure 3 lies between 19° 10' 1.6" to 21° 16' 29.75" North Latitude and 74° 43' 44.83" to 76° 53' 42.79" East Longitude. It has mixture of land covers i.e. road, water body, buildings, bare farms and largest area has sugarcane crop. This region has majorly agricultural land and a large number of fields have sugarcane crop at various growth stages. So, the focus of this study is to identify sugarcane at ripening/ growing and harvest/senescence stage along with other land covers. The study area has sugarcane crop at various growth stages. Fields of the study area are irregular in shape and size. Fields vary from one acre to twenty acres and even larger. The satellite data from LISS-IV, considering size of the crop fields. As the spatial resolution is 5.8m for the bands green, red and near infrared (NIR). These bands are most commonly used for identification of crops.

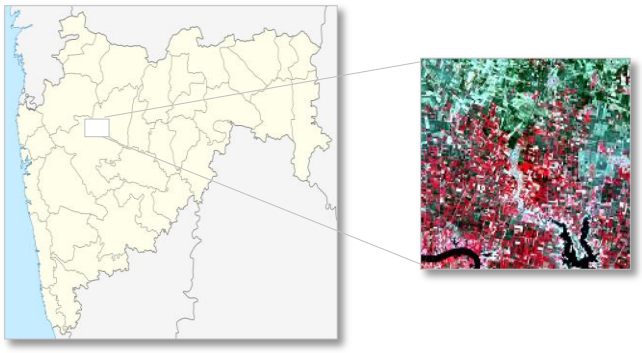

Figure 3. The study area is in the center of Maharashtra, Navin Kaigaon, India. Left hand-side is map of the Maharashtra state and right-hand side is image subset obtained from LISS-IV.

To construct the labelled dataset for image classification, satellite image of IRS LISS-IV sensor, high resolution imagery from commercial satellite was acquired for the study region. It has pixel size of 5.8m and spectral range from 0.52 μm to 0.86 μm. The image is color- Infrared image i.e. bands are combined as NIR, Red and Green (CI Image). Then training/ test dataset is created as image patches where regions of interest for every class are created separately. These image patches are further resized to desired patch size of 32x32, which is given as input to CNN's first layer. Dataset for six land covers (classes) have been prepared and manually checked as per the given number of image patches for every class:

1. Sugarcane Crop- Full Growth stage (6 to 8 months old) : 550 image patches
2. Sugarcane Crop- Harvest stage (12 to 14 months old) : 550 image patches
3. Water-body : 400 image patches
4. Settlement : 500 image patches
5. Road : 200 image patches
6. Bare-farm :300 image patches

Number of image patches are varying due to percentage of presence of land covers present in the study scene. Image patches are resized to 32x32 pixels size. This dataset is further increased to 3000 images per class using data augmentation methods like image rotation, flipping, Gaussian filtering for improving accuracy and reducing over-fitting. Data augmentation is transforming an image that doesn't change the image label. There are many ways to do it like rotation, scaling, flipping, cropping (random), color jittering etc. Also RGB intensities can be altered. Total images used are 3000x6 = 18,000, out of which 80% are used for training and 20% for testing purpose. Figure 4 shows sample image patches of the dataset created. The mean spectral reflectance curve of all six classes which is used to consider image patches for inclusion into the dataset is recorded while image patches were prepared to be added to final dataset. Those image patches whose mean reflectance values deviated outside the desired range were removed from the dataset, as they contained large number of mixed pixels.

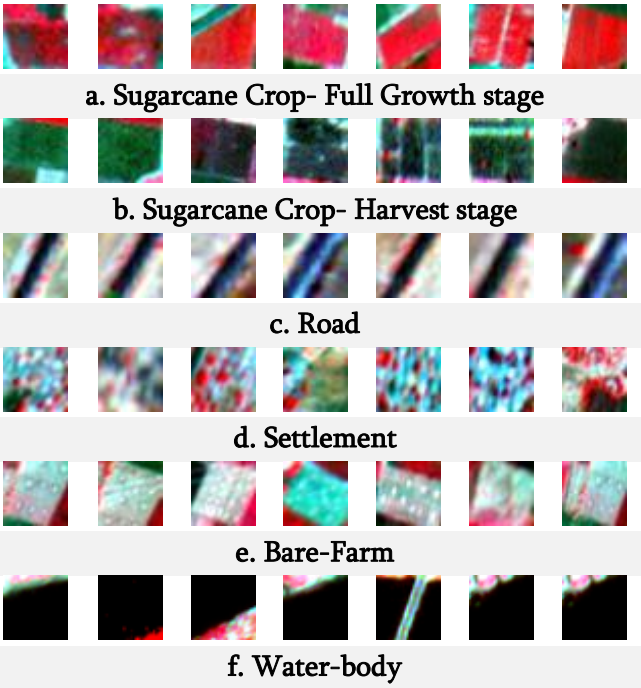

Figure 4(a-f): Sample image patches created for training and testing dataset, representing all six classes: Sugarcane Crop- Full Growth stage, Sugarcane Crop- Harvest stage, Road, Settlement, Bare-Farm and Water-body

Another labelled dataset consisting of image patches from Landsat8, using only PAN Band for image classification. Ten images are acquired for a period of one year, 19th April, 2016 to 24th May 2017 downloaded from the USGS Earth Explorer database

(United States Geological Survey) (http://earthexplorer.usgs.gov) [20]. Images considered for study have cloud cover less than 10%. Landsat-8 has 16 days repeat cycle, providing data at swath of 185kms. Images are Level 1T (terrain corrected) scene of the OLI/TIRS sensor, of path / row: 146 / 46. Landsat-8 provide images in 11 bands with spatial resolution of 30 m for seven multispectral bands, 15m for PAN band and 2 thermal bands acquired at 100m, resampled to 30m. Spectral resolution of PAN band is 0.50 μm to 0.68 μm. Then training/ test dataset is created as image patches where regions of interest for every class are created separately. These image patches are further resized to desired patch size of 32x32, which is given as input to CNN's first layer. Dataset for three land covers (classes) have been prepared and manually checked as per the given number of image patches for every class: Crop-farms: 350 image patches, Water-body: 350 image patches and Settlement: 300 image patches. This dataset is further increased to 1000 images per class using data augmentation methods, creating a total of 3000 image patches.

C. Experimental setup with parameters set for training network

We have implemented DCNNs using MatConvNet [19]. It is a MATLAB toolbox implementing Convolutional Neural Networks (CNN). CNNs need a lot of training data for learning process and also requires efficient implementations. MatConvNet provides this as it has methods for optimizations and supporting computations on GPUs. Building blocks of CNNs, convolution, normalisation and pooling can be easily combined and extended build DCNN models. MatConvNet is open-source released under a BSD-like license, simple to install and easy to use. Our model is implemented in MATLAB R2017a on Intel i7 -7500U CPU @ 2.70 GHz, NVIDIA GeForce 940MX graphic device with 8G byte graphic memory having Windows 10 operating system installed.

After creating DCNN models, they are trained by providing the created labelled datasets separately with LISSIV for five/six classes and Landsat-8, PAN dataset for three classes. The base learning rate is 0.0001, parameters to compute increments are: momentum = 0.9, and weight decay = 0.0005. The number of epochs are varied from 100 to 500 to check accuracy. DCNN models are trained with different batch sizes of 64, 128 and 256 where 128 is found to give best performance. Batch sizes is number of samples loaded into memory for the training phase of the DCNN. Models processes the complete training dataset, by making increments defined as batch size. Batch size is used for efficient computations and is also dependent on the hardware where CNN is trained. During training the DCNN, data used from the training set will minimize the error. The validation data is used to check the response of the CNN model on new and similar images, which network hasn't seen before i.e. it is not trained on. Validation or test data passes only in forward pass, as no error is calculated in this pass. As the training and testing process is completed, CNN model is saved and used to compute confusion matrix. Setting the value of learning rate is an important step as this values takes the network towards convergence, and selecting the appropriate value is an empirical process. Throughout the training phase of the CNN, the network generates three plots showing, Top1 error, Top 5 error, and objective for every successful epoch. The top1 error depicts that, the class with the highest probability is the true correct target, i.e. network found the target class. The top 5 error depicts that, the true target is one of the five top probabilities. The last layer, i.e. softmax is attached for final classification and it is fully connected. It has a filter depth of C i.e. number of classes of the remote sensed scene database. In our CNN model, filters of size 3x3xC, 5x5xC have been used with random weight initialization, where C is number of bands. To improve accuracy of our designed model, we incorporated data augmentation in the dataset preparation process.

IV. RESULTS AND DISCUSSIONS

To evaluate classification accuracy of the DCNN models, confusion matrix is generated after training process is completed. The impact of change in number of epochs on the classification accuracy is recorded. It has been observed that training time increases as the number of filters increase. Impact of different activation functions, ReLU and sigmoid is also computed and found that ReLU achieves higher classification accuracy. It is recommended to do extensive searching by applying a range of values to hyper-parameters to reach the best performance of the CNN models. This process requires a large amount of computations and is done using trial and error. We considered different size of filters in the range 1x1, 3x3, 5x5, 7x7 and also number of filters from 10 to 100, before making the final architecture.

We have evaluated performance of DCNNs model by considering "Loss", "Overall Accuracy", "precision" and "recall". The term "Loss" is used during the training process to find the appropriate hyper-parameter values for the model i.e. weight values. This value is continually optimized in the training process by updating weights. Overall accuracy is calculated after the loss value has been optimized. It measures the extent of how accurate is the model's prediction as compared to the labelled or target class. MatConvNet generates training and validation log likelihoods after every epoch during training cycle. Initially the curve of the validation log likelihood shows higher correct prediction values as the dataset of validation/ test is relatively 20% of the total dataset. Also, the distance between training and validation curves remains moderately constant as the training cycle proceeds depicting that there is very less over-fitting.

The performance of model is good as long as the cost curve for training and validation is reducing. It can be checked after every epoch. In case if it starts increasing means the model has started to over-fit and further training the model is of no use. The other measures used to compute performance of classification done by DCNN is precision, recall and kappa statistic. The precision is the fraction of predicted C instances which are true C instances. And the recall of a set of predictions is the fraction of true C instances that were correctly detected. These indexes are calculated from the confusion matrix C. The Kappa statistic compares the Observed Accuracy with Expected Accuracy. It is a measure of how closely the instances classified by the classifier model matched with the labeled data. Kappa value can also be used to compare performance of two classifiers performing the same classification work. It is recommended to use kappa value for classifiers made and evaluated on data sets with varying class distributions. Average Speed of the DCNN models were 4231 Hz means it could process 4231 images per second.

DCNN models 1 and 2 were trained and tested using LISSIV dataset for all six classes and it was observed from confusion matrix that class bare-farm shows least number of correct predicted instances. It is observed from image patches given as input, that bare-farms couldn't be marked correctly due to mixed pixels with leftovers of last crop. So models were again trained using LISSIV dataset for five classes i.e. Sugarcane Crop- Full Growth stage, Sugarcane Crop- Harvest stage, Road, Water-body and Settlement. The classification accuracy of model1 for six classes is 92.92% and for five classes (leaving bare-farm) is 97.56%. It was also observed that predicted labels for Sugarcane Crop- Full Growth stage and Sugarcane Crop- Harvest stage in both experiments with six and five classes is observed to be 100% correct by Model 1. This indicates that DCNN models are able to accurately identify crop present at different growth stages. Table 1 shows the various measures computed from confusion matrix for both the models trained and evaluated on five classes for different number of epochs. The batch-

size is kept to be 128 for all the experiments recorded in this table.

DCNN models were also trained by changing activation function to sigmoid and it was observed test accuracy of model 2 goes down drastically to 44% whereas for model1 is 92.75%. Since, we created a PAN dataset using images from Landsat8 sensor to check how our DCNN models perform on PAN data, it was observed that test accuracy is 66%. Fu et al., have used fully convolutional network for classifying high spatial resolution remote sensing imagery with 12 classes [17]. Accuracies of our models is recorded to be better than their approach. Authors have explained their results are less accurate due to confusing classes present in the images used and in their study they have used 12 classes and many of these classes (building, cement ground, road, and parking lot) potentially contains mixed pixels. In our study we have classified six classes and are found to be separable except for bare-farm. We have also evaluated performance of proposed models for two benchmarked datasets:

1. EuroSAT, a dataset created using Sentinel-2 satellite images, provided in 3 bands and 13 bands for 10 classes. Every class has 3000 images. We have used 3 band dataset for evaluating our models [6].
2. UC Mercedes consists of 21 class land use image dataset, having 100 images for each class. Each image is provided as 256x256 patch [22].

As depicted by table 2, our proposed DCNN models 1& 2 have overall accuracy of 98.1% & 96.28% on the LISS-IV dataset created. This classification accuracy is at par with accuracy attained on the two benchmarked dataset used. We have also compared the accuracy attained by Helber et al., on EuroSAT dataset for CIR image, since our dataset is in CIR band combination. DCNN model 1 has also attained same accuracy on the same dataset [6]. Performance is more efficient on our model since number of convolutional layers used is 3 & 4 and they have used fine-tuned ResNet-50 having 8 convolutional layers. PAN dataset trained on the same models doesn't have required accuracy.

Table 1. Overall Accuracy (OA), Kappa value, precision and recall measures for model 1 and 2.

	Epochs	OA %	Kappa	Precision (Producers Accuracy) %					Recall (Users Accuracy) %				
				G	H	R	S	W	G	H	R	S	W
Model 1	400	98.10	0.97	100.00	100.00	97.22	98.33	91.67	97.56	91.60	99.72	99.72	99.40
	300	97.28	0.97	100.00	100.00	98.33	96.67	91.39	98.09	92.07	98.33	99.15	99.40
Model 2	400	96.28	0.95	98.61	98.33	96.94	95.83	91.67	93.92	93.40	96.87	98.29	97.35
	300	96.83	0.96	98.89	96.39	98.61	98.06	92.22	97.80	93.03	98.61	97.78	97.08

(G: Sugarcane crop- Full Growth stage, H: Sugarcane crop- Harvest stage, R: Road, S: Settlement, W: Water)

Table 2. Validation/test accuracy results for four datasets.

DCNN Model	No. of Conv. layers	EuroSAT	UC Mercedes	**LISSIV**	**Landsat8 PAN**
Dataset Type		Multi-Spectral	Multi-Spectral	Multi-Spectral	PAN
Spatial Resolution		10m	30cm	5.8m	15m
DCNN model 1	**3**	97.00%	95.40%	**98.10%**	66.10%
DCNN model 2	**4**	95.20%	94.60%	**96.28%**	50.03%
ResNet-50 (22)	8	98.32%	96.42%	--	--

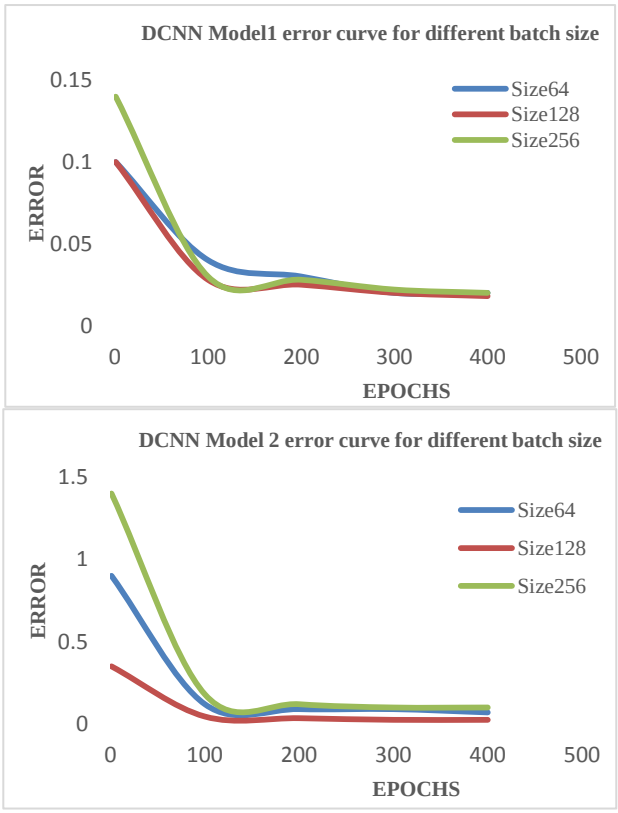

Figure 5. Error curve for DCNN Model 1 and 2 for batch size of 64, 128 and 256.

Impact of different size of batch provided (64, 128 and 256) on the performance of the DCNN Model 1 and 2 is as shown in figure 5, for LISS-IV dataset. Error reduces smoothly when batch size is 128 as compared to 64 even though the accuracy attained is same. For batch size of 256, there is slow convergence and better accuracy is obtained using either 128 or 64 batch size.

V. CONCLUSIONS WITH FUTURE SCOPE

This paper has presented dataset consisting of multispectral, high-resolution images from LISS-IV sensor and another dataset of PAN band created using coarse-resolution images from Landsat8 sensor. The dataset created using LISS-IV has shown promising accuracies on the two DCNNs models proposed here. The results show that DCNN models designed can identify crop land present in a satellite image along with commonly found land cover classes, water-body, bare-farm, road and settlement. As DCNN models are also evaluated for their performance using two similar publicly available benchmarked datasets, indicating that construction of models using described size of filters, number of filters and number of layers is suitable for prediction of land cover. The approach used in our models is patch-based and size of input patches taken as 32x32 is also justified, considering spatial resolution of LISS-IV and Landsat-8. Using only a specific set of features like spectral reflectance values or manually derived features, cannot help achieve good classification results. The proposed models are taking input as 3-D image tensor i.e. it considers the spectral features (3 bands) as well as spatial neighbourhood also (32x32). The models presented are working with 98% accuracy to identify or predict unknown image patches. This work can be further extended by using the trained DCNN models for large size satellite image classification.This LISS-IV dataset can be augmented using LISS-III images as it has same spectral resolution and different spatial resolution (23.5m). LISS-III images can be up-sampled (putting more pixels) to the resolution of LISS-IV (5.8m) before creating the image patches for dataset. Also more classes like other crops grown can be added by collecting images from different parts of the country.

VI. REFERENCES

[1]. Krizhevsky A., Sutskever I., Hinton G.E. ImageNet classification with deep convolutional neural networks. Neural Information Processing Systems. 2012 pp. 1097–1105.

[2]. Castelluccio M., Poggi G., Sansone C., Verdoliva L. Land Use Classification in Remote Sensing Images by Convolutional Neural Networks. 2016 Accessed online: http://arxiv.org/abs/1508.00092.

[3]. Hu F., Xia G.S, Hu J., Zhang L. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. 2015. Remote Sensing. 7, pp. 14680–14707.

[4]. Hubel, David H., Torsten N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. 1968. The Journal of physiology 195.1 pp. 215-243.

[5]. Yang Y. and Newsam S. Bag-of-visual-words and spatial extensions for land-use classification. 2010. Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems, pages 270-279. ACM.

[6]. Dataset] Helber P., Bischke B., Dengel A., Borth D. EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. 2017. http://arXiv:1709.00029v1 cs.CV]. (Accessed on 31 Oct 2017). (http:// http://madm.dfki.de/downloads).

[7]. Li H., Chao T., Zhixiang Wu, Jie Chen, Jianya Gong, Min Deng. RSI-CB: A Large-Scale Remote Sensing Image Classification Benchmark via Crowdsource Data. 2017. https://arxiv.org/abs/1705.10450 (Accessed on 7th Dec 2017).

[8]. Sheng G., Yang W., Xu T., Sun H. 2012. High-resolution satellite scene classification using a sparse coding based multiple feature combination. International Journal of Remote Sensing. Vol. 33, no. 8, pp. 2395–2412.

[9]. Zhao L., Tang P., Huo L. 2016. Feature significance-based multibag-of-visual-words model for remote sensing image scene classification. Journal of Applied Remote Sensing. Vol. 10, no. 3, p. 035004.

[10]. Zou Q., Ni L., Zhang T., Wang Q. 2015. Deep learning based feature selection for remote sensing scene classification. IEEE Geoscience & Remote Sensing Letters., vol. 12, no. 11, pp. 2321–2325, Nov. 2015.

[11]. Cheng G., Han J., Lu X. 2017. Remote Sensing Image Scene Classification: Benchmark and State of the Art. doi: 10.1109/JPROC.2017.2675998.

[12]. Langkvist, M., Kiselev, A., Alirezaie, M. & Loutfi, A. 2016. Classification and Segmentation of Satellite Orthoimagery Using Convolutional Neural Networks. Remote Sensing, Vol 8(329); doi:10.3390/rs8040329.

[13]. Basu S., Ganguly S., Mukhopadhyay S., DiBiano R., Karki M., Nemani R. 2015. Deepsat: a learning framework for satellite imagery. Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, p. 37. ACM.

[14]. Michael A. Nielsen. Neural networks and deep learning. 2017 (Accessed online: http:// neuralnetworksanddeeplearning.com on 30th Nov 2017)

[15]. Goodfellow I., Bengio Y., Courville A. Deep learning. 2017. MIT press book. Accessed online: http://www.deeplearningbook.org.

[16]. Nair V., Hinton G.E. 2010. Rectified linear units improve restricted boltzmann machines. International Conference on Machine Learning.

[17]. Fu G., Liu G., Zhou R., Sun T., Zhang Q. 2017. Classification for High Resolution Remote Sensing Imagery Using a Fully Convolutional Network. Remote Sensing 9, 498, doi:10.3390/rs9050498.

[18]. Nguyen T., Han J., Park. 2013. D.C. Satellite image classification using convolutional learning. Proceedings of the AIP Conference, Albuquerque, NM, USA, pp. 2237–2240.

[19]. Vedaldi A., Lenc K. 2016. MatConvNet Convolutional Neural Networks for MATLAB. http://arXiv: 1412.4564v3 cs.CV] 5 May 2016.

[20]. [Dataset], Landsat8. http://earthexplorer.usgs.gov. (Accessed on 30th May 2017).

[21]. [Dataset] WHU-RS Dataset. (http://www.tsi.enst.fr/~xia/satellite_image_project. html) (Accessed on 15th Oct 2017).

[22]. [Dataset] Yang Y. and Newsam S. 2010. Bag-Of-Visual-Words and Spatial Extensions for Land-Use Classification. ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM GIS), 2010. (http://vision.ucmerced.edu/datasets/landuse.html) (Accessed on 15th Oct 2017).

[23]. [Dataset] Brazilian Coffee Scenes Dataset. (www.patreo.dcc.ufmg.br/downloads/brazilian-coffee-dataset) (Accessed on 17th Oct 2017).

[24]. [Dataset] NWPU-RESISC45. (http://www.escience.cn/people/JunweiHan/NWPU -RESISC45.html.). (Accessed on 19th Oct 2017).