Land Cover Classification from Satellite Data using Machine Learning Techniques

Nisarg Vora[a], Arush Patel[b], Kathan Shah[b], Pallabi Saikia[b]
f20180254@pilani.bits-pilani.ac.in, {arush.pce18, kathan.sce18, pallabi.saikia}@sot.pdpu.ac.in
[a]*Birla Institute of Technology and Science, Pilani, India*
[b]*Pandit Deendayal Petroleum University, Gandhinagar, India*

***Abstract*—This work attempts automatic land cover classification of different parts of India into forest, built-up, agricultural land and water bodies using temporal remote sensing data. Data from Agra district, Uttar Pradesh has been used to train different models - k-nearest neighbours, decision trees, support vector machines and convolutional neural networks. These models are then tested in Ahmedabad and Gandhinagar, Gujarat. Google Earth Engine has been used to obtain data from Landsat 8 satellite images. For the purpose of classification, Normalized Difference Vegetation Index (NDVI) values are calculated by masking all other light bands except near-infrared and red light bands. Temporal images with NDVI labels are fed as input to train the models and subsequently, the performance of these models is compared. A convolutional neural network based on the U-Net architecture is found to produce the most accurate results, improving upon traditional machine learning techniques. The models implemented can be used to produce land cover maps for any region, with good accuracy, which can then be used for various applications like natural resource management, urban expansion etc.**

***Index Terms*—Remote sensing, Land cover classification, Machine learning, U-Net Architecture.**

I. Introduction

The method of obtaining data regarding the Earth's surface by the use of satellites instead of physical examination is known as Remote Sensing [13]. The application of Remote Sensing is stretched across many fields which mainly include geography [3], geology [3] and many different earth science subjects. Among different applications of remote sensing, classifying and categorizing human activities on the landscape using satellite data is popular recently [2], as it helps in establishing an accurate analysis on a piece of land and helps in its further classification. This task falls under the problem domain of **Land Cover Classification**. Land cover classification data is of great importance in the applications [16] of Natural Resource Management, Wildlife habitat protection, Urban Expansion/encroachment, Damage delineation (Tornadoes, flooding, fire, volcanic) and many more. Field survey was the primary method [8] used for land cover classification until a few decades back. However, it is very inaccurate and time-consuming. With developments in technology, it has been replaced by remotely sensed imagery analysis [32] which is capable to provide accurate results and covers a wider range of survey area. Modern statistical and computing techniques like machine learning and deep learning minimize the human intervention in the task of analysis and classification, making it accurate and agile. Even though satellite data has become a very important parameter for judging land cover classification using Machine Learning and Deep Learning Techniques, but it can not be directly used to classify land cover into different classes such as quality classes, capability classes or grade, depending upon the characteristics of the land and/or its potential for agricultural use. Normalized Difference Vegetation Index (NDVI) [6], which is calculated by measuring the reflection of infrared and red light is considered as the most important and accurate parameter for land cover classification [6]. NDVI values are so widely used as they can indicate the density of green on an area of land. NDVI values basically ranges from -1.0 to 1.0 [6], the positive values are usually for plants and vegetation whereas the negative values usually indicate presence of snow, water or cloud cover. Values that are almost 0 or less than 0.1 are because of the presence of urban built ups rocks etc. Values between 0.2 and 0.3 are because of small plants and crops whereas big values close to 1 represent tropical areas, mainly jungles. These rough trends indicate the relevance of NDVI to the problem of land cover classification.

There are different image classification approaches applied for classification of satellite imagery, include, Pixel Based Approach [33], Object Based Approach [33], and Semantic Approach Remote sensing data work on a per-pixel basis is used in the majority of methods for image analysis [31]. However, with advances in remote sensing technology, the spatial resolution has become clearer than normal objects of interest, leading to an increase in within-class variability [11]. Different machine learning and deep learning techniques have been applied [9] in this domain for effective satellite imagery classification. In case of land cover classification, the objective is not to assign only a single class label to a given satellite image, but to divide the given image into segments, each corresponds to one class out of a given set of land cover classes which requires semantic segmentation of the image [30]. One popular approach proposed by Long et al. to resolve this problem was to design the neutral network as a combination of an encoder and a decoder [12], [14], [23]. Decoder perform up-sampling and produces the classification maps by a Fully Convolutional Networks (FCN), which uses transpose convolution. To combine activation with different spatial dimensions, they are first up-sampled individually as required using transpose convolutions and then summed up. The result is then up-sampled by a final layer, so that the output spatial dimensions match those of the input image. The addition of skip connections produces a finer and more detailed segmentation map [12], [14]. With the motivation of the above work, we applied here U-Net architecture, where FCN is used as its component for the effective classification of land class with pixel by pixel approach. In this paper, we also implemented different machine learning models of image classification, that include models such as **Support Vector Machines, Decision Trees and K - Nearest Neighbor and mainly U - Net Architecture**. We focus on U - Net Architecture as it a modern popular trend and produces the most precise outcomes from the list mentioned above. Our paper incorporates the pixel based approach which executes more accurate results. We also display and compare results derived from these models of imagery analysis, and give insight on some of the future trends in this area. The area of our study includes Agra and Mathura, Uttar Pradesh and Ahmedabad and Gandhinagar, Gujarat. In our work passive remote sensing data collected by Landsat 8 [24] satellite has been used. Few related works in this domain include:

1) Noi et al. [29] introduced three non-parametric classifiers, Random Forest (RF), k-Nearest Neighbor (kNN), and Support Vector Machine (SVM) for LCC and were reported as the foremost classifiers at producing high accuracies.
2) Evaluation and comparison of the performance of four machine learning classifiers—support vector machine (SVM), normal Bayes (NB), classification and regression tree (CART) and K nearest neighbor (KNN)—to classify very high resolution images, using an object-based classification procedure was done by Qian et al. [21] in 2015.
3) Ma et al. [15] identified some methods that may advance supervised object-based image classification such as, deep learning and type-2 fuzzy techniques may further improve classification accuracy.
4) The proposal of multiple criteria for assessing algorithms for this task. In addition to standard classification accuracy measures, DeFries et al. [7] proposed criteria to account for computational resources required by the algorithms, stability of the algorithms, and robustness to noise in the training data.
5) Previous studies conclude that the RF algorithm is the most accurate machine-learning classifier, among the examined algorithms (i.e random forest (RF), support vector machine (SVM) and artificial neural network (ANN)) although it is necessary to further test the RF algorithm in different morphoclimatic conditions in the future.[28]

II. Data Description

Two sets of data have been considered for our study Landsat 8 and Copernicus Global Land Cover data set. The details and the characteristics of the data set are illustrated below:

Landsat 8: Time series data for NDVI was collected from the Landsat 8 Collection 1 Tier 1 8-Day NDVI Composite product. The data is collected by Landsat 8 - an American Earth Observation satellite launched in 2013 under Landsat [24], [27], a joint program of the United States Geological Survey (USGS) and National Aeronautics and Space Administration (NASA). The satellite images the entire Earth's surface at a spatial resolution of 30m, once every two weeks and records multi-spectral and thermal data. The Operational Land Imager (OLI) collects passive remote sensing data from nine spectral bands including the Near Infrared and Visible bands which we intend to use for the paper. The data has been atmospherically corrected in addition to standard geometric adjustments and georeferencing. Information from the thermal infrared sensor is also used to mask clouds. The imagery, hence, provides a clear view of land. The high temporal resolution of 8 days makes the data set suitable for the purpose of our work.

Copernicus Global Land Cover data set: A chain of bio-geophysical products examining the condition and development of the global surface is being produced by the Copernicus Global Land Service (CGLS) - LC100 collection 2 which is also regarded as an element that controls a multi-motive Land service. Derived from the PROBA-V 100m time series data for the year of 2015, this dynamic land cover map at 100m spatial resolution issues a primary land cover model (scheme). Continuous field surface layers for all primary land cover types that supply corresponding approximations for ground-vegetation cover are also included in this product. [10].

Study Areas: The study area comprises of the data from different regions of India. Images of Agra, Mathura and the surrounding region of Uttar Pradesh were used for training the classifier for land class classification whereas images of Ahmedabad and Gandhinagar, Gujarat were used for testing the performance of the developed classifies. The chosen locations have the even distribution of different land cover classes, hence suitable for developing an unbiased classifier. Different views of the survey location of Agra is shown in Figure 1.

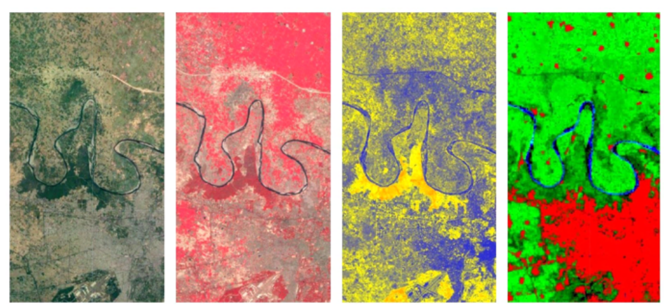


Fig. 1: Different views of the training location - Agra. (1) True colour satellite image (2) False colour infrared image (3) Map of NDVI values (4) Ground truth land cover [24].

III. Methodology

The satellite image data is not suitable to be applied directly for classification of land cover. It need to be passes through different preprocessing approaches to make the data suitable for land cover classification. The proposed methodology for the land cover classification is provided in Figure 2. It consists of data collection and preparation; Data Preprocessing; Modelling and then performance analysis.

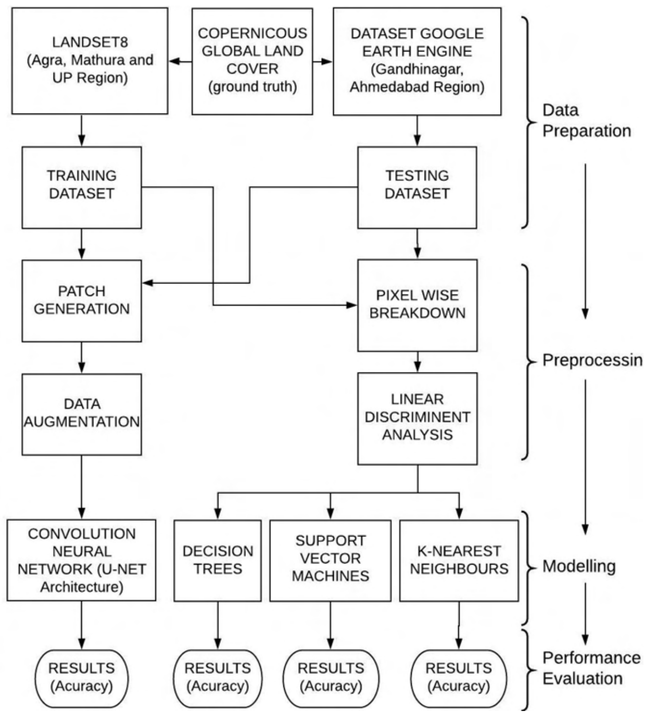


Fig. 2: Workflow of the proposed methodology

A. Data Preparation

NDVI time series data: The data was collected and prepared using Google Earth Engine Javascript API [10]. Google Earth Engine allows users to carry out geo-spatial analysis on Google's cloud infrastructure. The client libraries provide Python and JavaScript wrappers around the web Application Programming Interface (API). In our work, the JavaScript API was used to visualize, prepare, process, and export satellite imagery and data. The Landsat 8 NDVI Composite product was first imported as an Image Collection (a class defined in the API, representing a set of images). The extents of the images were constrained to the required region - Agra and Mathura. The collection was then filtered for bimonthly periods (January to February, March to April, and so on). While a higher temporal resolution is desirable, it would not be possible to mask all cloudy regions using the limited number of images in a smaller duration. After filtering, the median NDVI value was computed for every pixel in each bimonthly period. The six images thus achieved were stacked together, producing a single six-band image.

Ground truth land cover data: The Copernicus Global Land Cover data set has different "cover fraction" bands which provide proportional estimates for different land cover classes. Appropriate thresholds were estimated for each of these bands through an iterative process of comparison with land cover maps available via ISRO's Bhuvan platform. The masks obtained by applying these thresholds were concatenated to arrive at a mutually exclusive and exhaustive classification of the study area. The maps obtained were found to have satisfactory accuracy but suffered from the low spatial resolution of the original data set - 100m. The work aims to produce finer land cover maps by operating on satellite imagery with a spatial resolution of 30m. The land cover classes used are: **Built-up land, including urban and rural areas, Cultivated and managed vegetation, Water Bodies** and **Forest land**.

B. Data Preprocessing

Different Preprocessing techniques employed for the processing of the data to make it suitable for modelling.The description of different proprocessing approaches applied on the data are illustrated below:

Linear Discriminant Analysis: The use of Linear Discriminant Analysis (LDA) [18] are broadly equipped in numerical databases and machine learning to discover linear patterns which differentiates classes or events. It attempts to find linear combinations of variables that explain the data. LDA tries to maximize the separation between the classes.

Linear discriminant analysis works as a class-wise discrimination and dimensionality reduction technique for this work which drastically reduced the processing time and made results more accurate. Since NDVI values were calculated bi-monthly, the data initially had 6 features i.e. 6 dimensional. As the data points were classified into 4 different classes, LDA reduced the number of features to 3 i.e. 3 dimensional.

Patch generation: Since, the convolutional neural network we implemented expects inputs to have dimensions of 256×256, the satellite images extracted cannot be used directly. To this end, a patch generation system was implemented. The system takes images of any width and height as input and produces cropped image patches of the required dimensions using a given stride. The patches obtained cover the input image completely, irrespective of its dimensions.

Data Augmentation: A huge quantity of training analysis is required by deep neural networks. It may not be possible for the network to perform well unless the training data provides a representative sample of the distribution of different types of land cover in different areas. Therefore, to augment the size of the training dataset, every patch was rotated by 90◦ thrice. This allowed us to train the network on four times as much data.

C. Modelling: Machine Learning Techniques

Once, we pre-processed the data, it can be directly applied to machine learning model for effective classification of land cover. The aim of applying machine learning models for modelling is to achieve automatic land cover classification. To do so, we have considered different popular machine learning models to compare their performances and consider the best out of them. The task of machine learning which maps an input to the output on the basis of the training data is known as Supervised learning [17], [26]. The training data is a group of input-output example pairs used to train the classifier. If enough data is available then the trained model usually identifies the underlying data patterns and is then able to make predictions on previously unknown input. We have considered few popular supervised machine learning techniques to identify the patterns within the data. Overview, advantages and drawbacks of different models of the techniques for imagery analysis are discussed below.

*1) **Support Vector Machines**:* A Support Vector Machine (SVM) [19] is a distinctive classifier officially defined by a separating hyperplane. This means, the optimal hyperplane which leads to the categorization of latest examples can be achieved by the output of the algorithm, if the labeled training data (supervised learning) is provided. The work here utilized soft margin SVMs keeping in consideration that hard margin support vector machines are very sensitive to outliers and may deteriorate the model performance.

*2) **Decision Trees**:* Decision trees [22] are the implementation of a tree like pattern of decision making and support device, which comprises the event consequences while taking the price and the utilities under consideration. Decision trees are usually applied in decision analytics where they can obtain a strategy out of the possible options which can be most promising to reach the desired goal.

*3) **K - Nearest Neighbor**:* The K-Nearest Neighbor (KNN) algorithm works on the principle that the existence of similar objects are closer as to the dissimilar. Training process in KNN is very efficient as only the training points and their labels need to be stored. The number of classes K, to be used for classification can be chosen by the user. This algorithm represents similarity by a simple math concept i.e. the displacement between the points on a graph. K-Nearest Neighbors can be a very accurate method to classify the pixels. Over a period of time, pixels belonging to the same classes tend to have similar features and hence they are closer. It classifies a point to a class in which the maximum of its closest neighbors belong to. KNN can be developed on large data set unlike other classifiers like SVM.

*4) **U - Net Architecture**:* The architecture was proposed by Ronneberger et al.[23], that builds upon and extends the FCN architecture for more precise segmentation on a smaller training dataset. The U-Net architecture can be divided into three major parts: The down-sampling path; The bottleneck; and The up-sampling path.

The **Down-sampling path** is made up of four identical blocks. Each block encloses a sequence of Convolutional layer ($Filter(f) = 3, Stride(s) = 1, Padding(p) = 0$), Convolutional layer ($f = 3, s = 1, p = 0$), Max Pooling layer ($f = 2, s = 2$).

At each down-sampling step (max-pooling), the number of channels is doubled, starting with 64 channels in the first block. The down-sampling path captures not only features relevant to classification, but also contextual information about the location of different segments.

The **Bottleneck path** consists of Convolutional layer ($f = 3, s = 1, p = 0$), Convolutional layer ($f = 3, s = 1, p = 0$)

The **Up-sampling path** is made up of four identical blocks. Each block encloses the following sequence of Up-sampling transpose convolution layer ($f = 2, s = 2$), Concatenation with appropriately cropped feature map from down-sampling path, Convolutional layer ($f = 3, s = 1, p = 0$), Convolutional layer ($f = 3, s = 1, p = 0$).

Modifications to U - Net Architecture

Some changes were made to the architecture of U-Net to make it more efficient (performance wise). A convolutional neural grid which is built considering the above design was implemented using TensorFlow in Python.

(a) Convolutional layers down-sample the image, leading to shrinkage of the image's spatial dimensions. This is particularly troublesome for skip-connections as concatenation of volumes at the time of up-sampling requires that the width and height of the volumes match. Using the same padding before applying the convolution operation prevents this down-sizing and solves the problem. This also enhances the symmetry in the network, with the output layer having the same spatial dimensions as the input image, reducing the overhead of image processing which would otherwise be required for training the network.

(b) As a neural network is trained, there is a continuous rotation and alteration in the input of each layer because the attributes of the former layers change. This decelerates the learning and necessitates careful parameter initialization. This problem of internal covariate shift can be addressed by normalizing layer inputs to a learnable mean and variance, stabilizing the distribution of the activations. Batch Normalization allows us to use higher learning rates and makes the network more robust to the choice of hyper-parameters. Additionally, it also has a regularizing effect and helps prevent overfitting.

(c) The shape of the image accepted as input by the network was fixed to 256 × 256. A patch size of 256 was found to provide sufficient spatial context to the network and fit well with the computational constraints.

(d) The model implemented has 31,051,332 parameters, of which 31,039,556 are trainable.

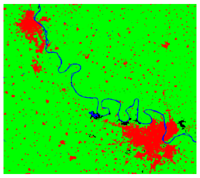
Loss Function: Since the neural network implemented by us uses softmax activation in the output layer, categorical cross entropy was used as the loss function. This is the natural choice for any multi-class classification problem where target labels are one-hot encoded [4]. The categorical cross entropy loss is computed as follows:

$$CE = -\log \frac{\exp s_p}{\Sigma_{C_j} \exp s_j}$$

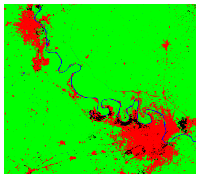
Optimizer: Adam (Adaptive Moment Estimation) [25] is an optimization algorithm designed specifically for deep neural networks that uses adaptive learning rates for each parameter. It computes exponentially weighted averages of gradients and squared gradients.

IV. Experimental Setup

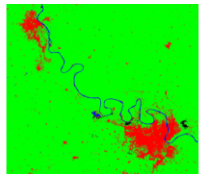
For developing and testing of ML models Google Colab [5] is a hosted Jupyter notebook service which provides a web-based development environment for writing and executing arbitrary Python code within a browser. Software like Scikit-learn [20] and TensorFlow [1] were used for implementing



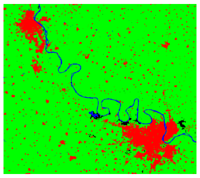
(a) Ground truth



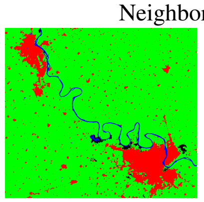
(b) Support Vector Machines



(c) K-Nearest Neighbors

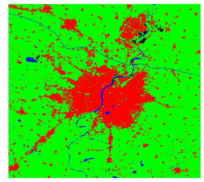


(d) Decision Tree

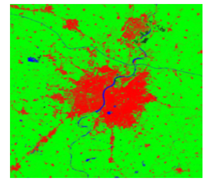


(e) U-Net

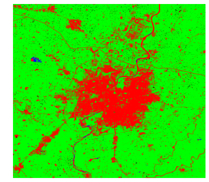
Fig. 3: Ground truth and predictions for the **training location - Agra and Mathura, Uttar Pradesh**.



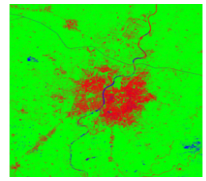
(a) Ground truth



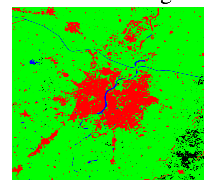
(b) Support Vector Machines



(c) K-Nearest Neighbors



(d) Decision Tree



(e) U-Net

Fig. 4: Ground truth and predictions for the **testing location - Ahmedabad and Gandhinagar, Gujarat**.

various models based on machine learning algorithms like SVM, k-nearest neighbours and decision tree classification, for data preprocessing and augmentation.

V. Results of Experiments

Four algorithms namely, Support Vector Machine (SVM), Decision Tree (DT), K-Nearest-Neighbour (KNN), CNN model with U-Net Architecture are considered for comparison. The performance of the algorithms on the dataset have been analysed by using Accuracy metric as we have the balanced set of data. The comparison results isprovided in Table I.

TABLE I: Performance Comparison Results of Modelling

Model	Training Accuracy	Testing Accuracy
K-Nearest Neighbours	93.52%	75.53%
Decision Trees	98.18%	82.71%
Support Vector Machines	89.63%	87.37%
CNN (U-Net architecture)	95.06%	89.31%

All four models i.e. SVMs, Decision tree, K-Nearest Neighbour and U-Net framework, were evaluated on the testing location Agra and Mathura, Uttar Pradesh (shown in Figure 3) and also Ahmedabad and Gandhinagar, Gujarat (shown in Figure 4). The region has sufficiently different characteristics from the training location to provide a good estimate of the quality of classification for any general region. Particularly, the seasonal river flow and large green areas in the cities make the region harder to classify correctly for any system. Categorical accuracy was chosen as the unit to measure performance of the models as it is the natural choice for a multi-class classification problem.

Support vector machine, compared to other classifiers, were found to be better at identifying forest areas. Certain urban areas like Fatehpur Sikri, Achhnera and Kiraoli were also very prominent in the output in case of SVM as found in field survey as well. This was found to be missing in most of the other classifiers. Support vector machines, however, miss-classified the small villages and towns as agricultural areas. Decision trees had very high training accuracy. However, they were found to have lower accuracy on the testing dataset, which indicates overfitting. K-nearest neighbors had the lowest testing accuracy among all the classifiers. Convolutional neural networks were best at identifying Keethal lake which was missed by other classifiers. Among all the classifiers, convolutional neural networks had the highest test accuracy and produced a much smoother output. This can be attributed to the fact that they take spatial variations directly into account, as opposed to the pixel-wise classification scheme used by the other models. Hence, we can conclude that out of all the classifiers, convolutional neural networks were best suited for the work of automatic land cover classification from temporal satellite images. The network was first trained with typical values for hyper-parameters on a small training set. Training the network for 70 epochs with a learning rate of 0.005 and default values for the other hyper-parameters was found to produce the best results. The convergence of the model is provided in Figure 5.

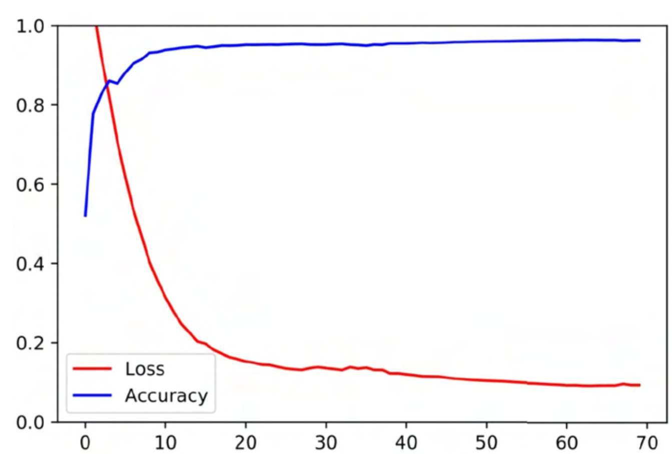


Fig. 5: Curves for loss and accuracy against number of epochs.

VI. Conclusion and Future Scope

The work explores the applicability of machine learning and deep learning techniques to the problem of land cover classification and lays out the advantages and disadvantages of different approaches. Traditional machine learning models like support vector machines, decision trees, etc. provide a method for pixel-wise classification based on learning from the temporal variations in the NDVI time series. On the other hand, Convolutional neural networks, as implemented in our project, make use of both spatial and temporal variations for segmentation. However, they also tend to require considerably more training data to produce good results. The U-Net architecture resolves this problem to a considerable extent, especially when trained with some custom modifications on an augmented dataset. The high accuracy attained with limited training is a testament to the success of skip-connection based architectures for semantic segmentation. Meanwhile, traditional machine learning techniques were also found to produce good results, highlighting their relevance to image classification tasks. The work aims to help in reducing the overheads in field survey method which is still being used in many parts of India.

The model has currently been trained on a small dataset consisting of satellite images of Agra and Mathura. Thus, the model has currently learnt the seasonal and crop variations in the Agra District area only. With proper training of seasonal and crop variations of an area, the model would be able to classify at other test locations as well with higher accuracy. In the current work, we considered only four classes of land cover- urban, rural, forest and water bodies. Models can also be extended to classify areas into other classes like wasteland, arid-land, rural areas etc. Expanding the scope of the work to more land cover classes should, hence, help improve performance. Some preprocessing techniques may be useful for artificially increasing the extent of the images to solve the problem of noisy classification of the image.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, and Craig Citro et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] James Richard Anderson. *A land use and land cover classification system for use with remote sensor data*, volume 964. US Government Printing Office, 1976.

[3] Ghassem Asrar and Ghassem Asra. Theory and applications of optical remote sensing. 1989.

[4] Raúl Gómez Bruballa. Understanding categorical cross-entropy loss, binary cross-entropy loss, softmax loss, logistic loss, focal loss and all those confusing names.

[5] Tiago Carneiro, Raul Victor Medeiros Da Nóbrega, Thiago Nepomuceno, Gui-Bin Bian, Victor Hugo C De Albuquerque, and Pedro Pedrosa Reboucas Filho. Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *IEEE Access*, 6:61677–61685, 2018.

[6] J Cihlar, L St Laurent, and JA Dyer. Relation between the normalized difference vegetation index and ecological variables. *Remote sensing of Environment*, 35(2-3):279–298, 1991.

[7] R.S DeFries and Jonathan Cheung-Wai Chan. Multiple criteria for evaluating machine learning algorithms for land cover classification from satellite data. *Remote Sensing of Environment*, 74(3):503–515, 2000.

[8] Giles M Foody. Status of land cover classification accuracy assessment. *Remote sensing of environment*, 80(1):185–201, 2002.

[9] Alba Gil, Q Yu, Agustín Lobo, P Lourenço, Luis Silva, and H Calado. Assessing the effectiveness of high resolution satellite imagery for vegetation mapping in small islands protected areas. *Journal of Coastal Research*, pages 1663–1667, 2011.

[10] Google. Google earth engine - copernicus global land cover.

[11] Manideep Kolla, Aniket Mandle, and Apoorva Kumar. Eye in the sky, 2018. GitHub Page.

[12] Fei-Fei Li, Justin Johnson, and Serena Yeung. cs231n - lecture slides, 2017.

[13] Chor Pang Lo. Applied remote sensing. 1986.

[14] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[15] Lei Ma, Manchun Li, Xiaoxue Ma, Liang Cheng, Peijun Du, and Yongxue Liu. A review of supervised object-based land-cover image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 130:277–293, 2017.

[16] GJ McDermid, RJ Hall, GA Sanchez-Azofeifa, SE Franklin, GB Stenhouse, T Kobliuk, and EF LeDrew. Remote sensing and forest inventory for wildlife habitat assessment. *Forest Ecology and Management*, 257(11):2262–2269, 2009.

[17] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. In *Foundations of Machine Learning,*, 2012.

[18] Shivaprakash Muruganandham. Modern applied statistics with s, 2016.

[19] Savan Patel. Chapter 2: Svm (support vector machines), 2017.

[20] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, and Olivier Grisel et al. Scikit-learn: Machine learning in python. *CoRR*, abs/1201.0490, 2012.

[21] Yuguo Qian, Weiqi Zhou, Jingli Yan, Weifeng Li, and Lijian Han. Comparing machine learning classifiers for object-based land cover classification using very high resolution imagery. *Remote Sensing*, 7(1):153–168, 2015.

[22] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[24] David P Roy, Michael A Wulder, Thomas R Loveland, Curtis E Woodcock, Richard G Allen, Martha C Anderson, Dennis Helder, James R Irons, David M Johnson, Robert Kennedy, et al. Landsat-8: Science and product vision for terrestrial global change research. *Remote sensing of Environment*, 145:154–172, 2014.

[25] Sebastian Ruder. An overview of gradient descent optimization algorithms.

[26] Stuart J. Russell and Peter Norvig. In *Artificial Intelligence: A Modern Approach, Third Edition*, 2010.

[27] United States Geological Survey. Landsat 8.

[28] Swapan Talukdar, Pankaj Singha, Susanta Mahato, Shahfahad, Swades Pal, Yuei-An Liou, and Atiqur Rahman. Land-use land-cover classification by machine learning classifiers for satellite observations—a review. *Remote Sensing*, 12(7), 2020.

[29] Phan Thanh Noi and Martin Kappas. Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using sentinel-2 imagery. *Sensors*, 18(1), 2018.

[30] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell. Understanding convolution for semantic segmentation. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 1451–1460. IEEE, 2018.

[31] Robert C Weih and Norman D Riggan. Object-based classification vs. pixel-based classification: Comparative importance of multi-resolution imagery. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(4):C7, 2010.

[32] Yichun Xie, Zongyao Sha, and Mei Yu. Remote sensing imagery in vegetation mapping: a review. *Journal of plant ecology*, 1(1):9–23, 2008.

[33] Gao Yan, J. -F. Mas, B. H. P. Maathuis, Zhang Xiangmin, and P. M. Van Dijk. Comparison of pixel-based and object-oriented image classification approaches—a case study in a coal fire area, wuda, inner mongolia, china. *International Journal of Remote Sensing*, 27(18):4039–4055, 2006.