International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V72I6P107
Volume 72 Issue 6, 64-73, June 2024

Original Article

Pixel-Based vs Patch-Based Classifiers for the LULC Classification

Keerti Kulkarni[1], Priyadarshini K Desai[2]

[1,2] Department of ECE, BNM Institute of Technology, Bangalore, India.

[1]Corresponding Author : keerti_p_kulkarni@yahoo.com

Received: 11 March 2024 Revised: 15 May 2024 Accepted: 24 May 2024 Published: 29 June 2024

Abstract *- A vast majority of the policy decisions by the civic authorities depend on the geography of the given area. The land use and the land cover of the area also influence the infrastructure facilities availed by the civic bodies. The problem with LULC mapping of the urban areas is the class imbalance. Very few existing algorithms take into consideration this class imbalance. The novelty of this work is the handling of this class imbalance at two levels: the data level and the classifier level. At the data level, uniformly sampled training samples and area-proportional training samples are considered and compared. At the classifier level, pixel-based and patch-based classifiers are considered and compared. A pixel-based Parametric Classifier (Maximum Likelihood Classifier), trained on uniformly sampled training samples, gives an overall accuracy of 73.21% and an overall accuracy of 75.81% when trained on area-proportional training samples. Pixel-based Non-parametric Classifier (multiclass Support Vector Machines), trained on area-proportional training samples, gives an overall accuracy of 83%. The study area is the Bangalore Urban District, and the remotely sensed images are from LANDSAT-8. Patch-based convolutional neural networks give a superior accuracy of 91.2%. Hence, for an imbalanced class dataset, a classifier-level approach (convolutional neural networks) works better, as they look at patches of images rather than individual pixels.*

Keywords *- Imbalanced dataset, Convolutional Neural Networks, Land Use Land Cover (LULC), Maximum Likelihood Classifier (MLC), Support Vector Machines (SVM).*

1. Introduction

Land Use Land Cover (LULC) has been traditionally classified from remotely sensed images. Multispectral classification using machine learning algorithms has received a lot of interest in the past decade. Two areas of interest play a major role in the organization of this work. The first one is the imbalanced dataset (images where we have a greater number of pixels for one class and a much smaller number of pixels for the other class). This kind of imbalance affects the accuracy of the algorithms. In this work, we show how we can overcome this imbalance to maximize the overall accuracy of classification. The objective of this work is to get maximum classifier accuracy for an imbalanced class dataset.

The class imbalance is handled at two levels, namely, the data level and the classifier level. In the data-level approach, the ratio of the training samples from each class is adjusted to achieve better accuracy at the classification stage. Two sets of training pixels are used for the data-level implementation. SET-1 is uniformly sampled, and SET-2 is sampled in an area-proportional manner. In the classifier-level approach, classifier parameters are tweaked to achieve better accuracy. For this purpose, three different classifiers are considered. Firstly, a traditional pixel-based, non-parametric Maximum Likelihood (ML) classifier is considered. This classifier is trained with both SET-1 and SET-2. These classifiers are simple to implement, but the parameters and accuracy are static, as the parameters required for the training are extracted from the training data itself. Next, a pixel-based, parametric Support Vector Machine (SVM) classifier is trained and implemented with only SET-2. In non-parametric SVM classifiers, the hyperparameters can be tweaked to obtain a greater accuracy. Both these classifiers work on individual pixels of the image. The classification accuracy can be improved if the classifiers consider the neighbours of the pixels also during the training phase. Hence, the patch-based Convolutional Neural Networks (CNN), which are context based, are implemented. In a patch-based CNN, the images are clipped such that each clip depicts an individual land cover class clearly. These clips are then fed to the network for training. For pixel-based classifiers, feature selection is an important step. The accuracy of the classifier depends on the feature selection. But for a patch-based CNN, the features are extracted in the network itself, hence, no external feature selection blocks are necessary. Four different classes, namely, water, vegetation, built-up and soil, have been identified for the classification. The choice of the number and the type of classes is decided by the geographic area under consideration.

2. Literature Survey

Most of the common machine learning classifiers are sensitive to class imbalance (Blagus & Lusa, 2010). Class imbalance can be divided into two types: between class imbalance and within class imbalance (Douzas & Bacao, 2018; Jo & Japkowicz, 2004). The former refers to the asymmetry between the two classes, whereas the latter refers to the fact that the spectral signatures vary even for the same class depending on the geographic area. The class imbalance problem can be overcome either by resampling or heuristic methods. A variety of approaches have been reviewed and proposed to deal with the class imbalance (Kaur et al., 2019; Fernández et al., 2013). The first is a cost-based approach, wherein a higher cost is attached to the minority classes, and a higher cost is attached to the majority classes. The second is the algorithm-based approach, where the specific classifiers are modified to improve the learning of the minority classes. The third is the resampling method, where the classes are balanced by either deleting the majority class instances or by generating more minority classes artificially. In the resampling methods, there are three different subgroup approaches: undersampling, oversampling and a hybrid approach, which is a combination of both undersampling and oversampling (Luengo et al., 2020). Random Oversampling (ROS) generates artificial instances randomly and is easier to implement (Sharififar et al., 2019). Hounkpatin et al., 2018 have shown that the ROS methods actually degrade the performance of the classifier as compared to when the dataset was imbalanced.

Random Undersampling randomly removes the instances belonging to the majority class, but this was found to be detrimental to the results (Feng et al., 2019). It was also found that the resampling methods give inconsistent results between various classifiers, such as RF and SVM (Maxwell et al., 2018). SMOTE (Chawla et al., 2002) was the first heuristic oversampling method and has been generally used to tackle the class imbalance in the LULC classification. A number of studies have implemented the SMOTE algorithm in the LULC classification domain and have shown better classification results (Jozdani et al., 2019; Bogner et al., 2018).

Authors have also proposed an adaptation of the SMOTE for deep learning approaches (Zhu et al., 2020). In spite of its popularity, the SMOTE algorithm suffers from noisy instances (due to random selection) (Douzas & Bacao, 2019). Cluster-based oversampling approaches solve this problem. Fonseca and Douzas, 2021 have proposed the use of the k-means SMOTE algorithm, in which the artificial instances are generated using two different methods, to eliminate the problem of noisy samples. A variety of pixel-based classifiers have been traditionally used for the LULC classification. The accuracy of the decision trees suffers because of the imbalanced classes (Panigrahi et al., 2021). Algorithm-based methods generally focus on non-ensemble-based classifiers like support vector machines (Shao et al., 2014).

For an imbalanced class, Parametric decision trees perform better with an overall accuracy of 93% (Pech-May et al., 2022), compared to ML classifier (Balha et al., 2021). The algorithm-based approaches are found to perform better than the resampling methods (Lee et al., 2016). The performance of these classifiers depends on a variety of parameters. A very important parameter is the sampling strategy of the training area. Colditz, 2015 has shown that the area-proportional strategy for allocation of the training samples achieves the maximum accuracy. Along with the classifier itself, features and training data have to be defined. Features can be selected based on their separability (Kulkarni K & Vijaya, 2021). There are various aspects of training data selection (Li et al., 2014; Radoux et al., 2014; Jin et al., 2014). A few studies recommend heterogeneous training data for the classification (Mishra et al., 2019; Deur et al., 2020), but for large geographical area mapping, a homogenous training area achieves better results (Blanco et al., 2013; Phinzi et al., 2023). [2-32]

The studied literature mainly handles the imbalance of the classes by changing the methodologies for sampling the training data. Comparison of different classifiers (both pixel-based and patch based) for an imbalanced dataset is rarely explored. In this work, the class imbalance is tackled at two levels. First, at the data level, where the training samples considered are chosen in an area-proportional manner and non-area-proportional (uniform) manner. Second, at the classifier level, pixel-based parametric and non-parametric classifiers and patch-based CNN classifiers are implemented. The main aim of this work is to explore various options which can be used classification of geographic areas with imbalanced classes.

3. Study Area and Materials

The study area is the Bangalore Urban District (Figure 1). Bangalore is the capital city of Karnataka, a southern state of India. The LANDSAT-8 images dated 1st March 2021 are downloaded from the USGS Global Visualization viewer (GloVis).

4. Methodology

The gist of the implementation is shown in Figure 2. Before tackling the imbalanced classes problem, the raw satellite images have to be corrected for atmospheric interference and clipped to the required district boundaries.

4.1. Data-Level Approach (Dataset Preparation)

The training dataset is 0.2% of the actual study area. This value is chosen in a trial-and-error manner, and it is found to be sufficient for the given study area. To sample area-proportional training pixels, the proportion of the area occupied by each of the LULC classes has to be anticipated. In short, a reference map (or data) which may give a rough idea about the proportion of the area occupied by each land cover class is needed.

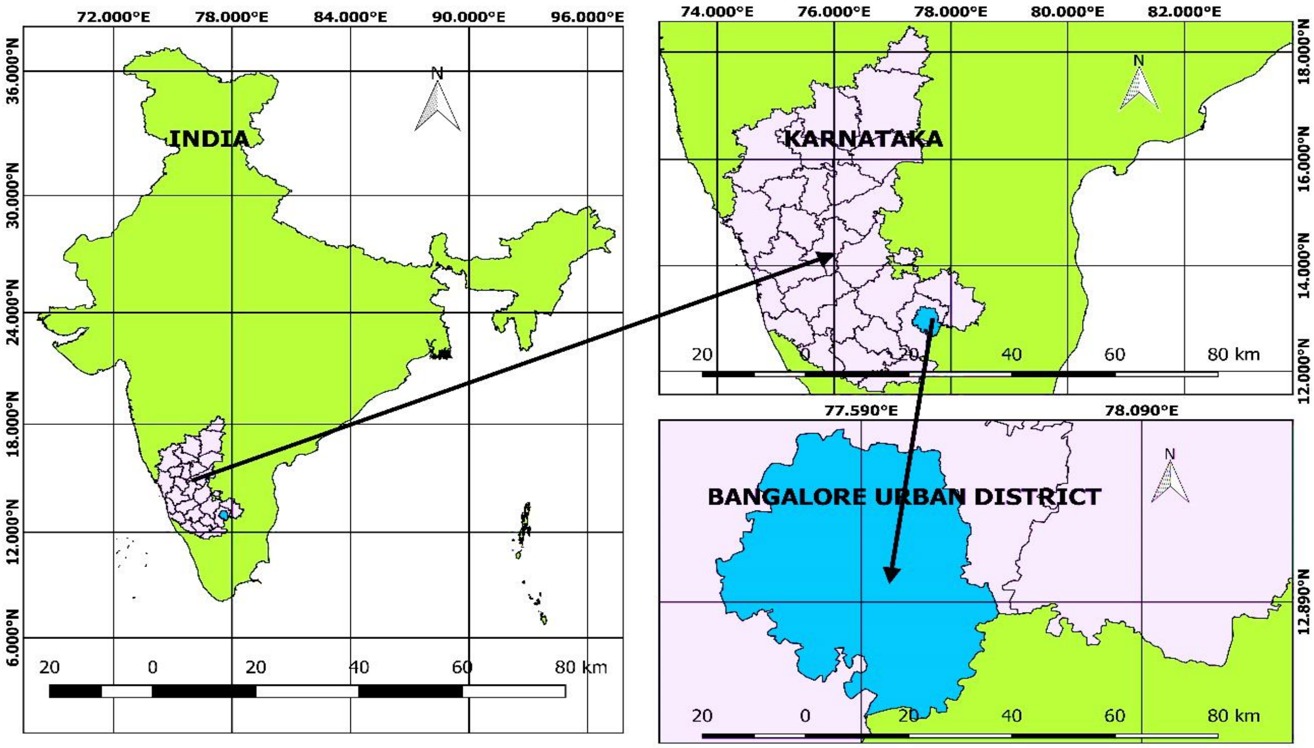

Fig. 1 Study area

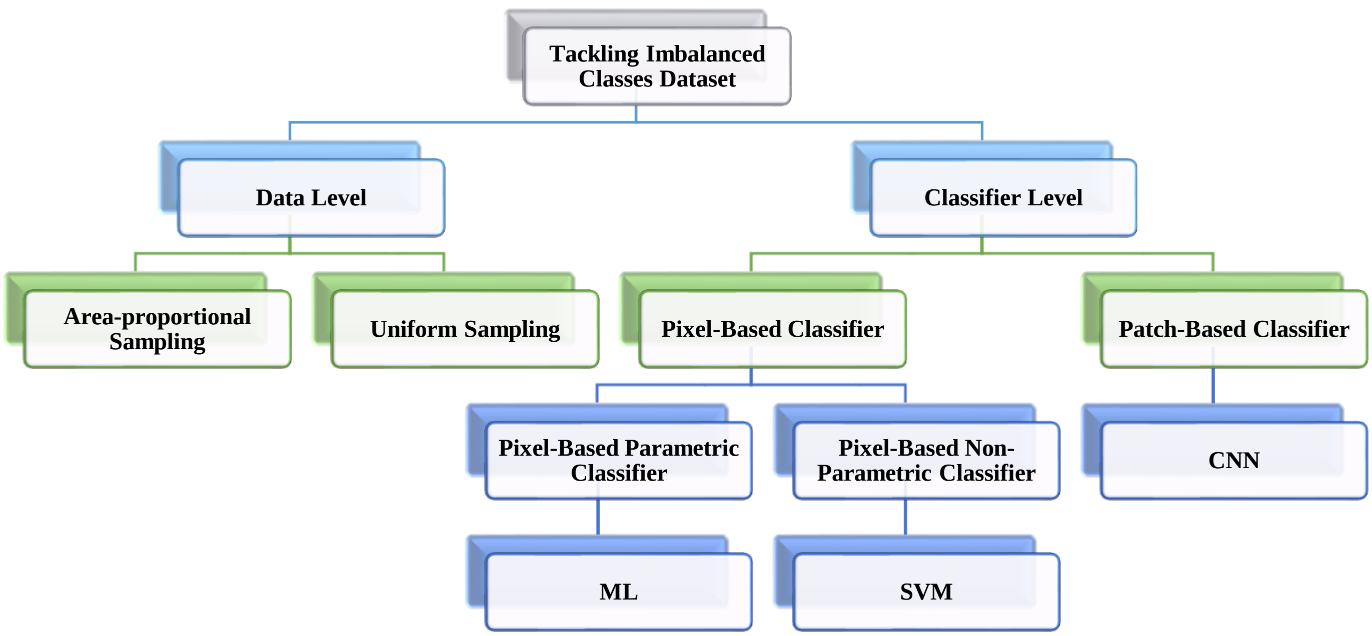

Fig. 2 Workflow of the proposed methodology

The LULC map for the year 2015 for the study area is available on the National Remote Sensing Centre (NRSC) website. The statistics in the reference data show that the built-up occupies around 45% of the study area, soil occupies 45%, water 2% and vegetation 8%. The values used here are rounded off so that calculating the proportional training samples becomes easier. Also, even though these values correspond to a different year, it still give a fair idea about the proportion of each of the LULC classes. Using this data, the training pixels are sampled in two ways. SET-1 consists of the uniformly sampled training pixels, whereas SET-2 consists of area-proportional training pixels. The approximate details of this are given in Table 1. The ML classifier and the SVM are trained with both SET-1 and SET-2.

Table 1. Training samples allocation mechanism

Class	% Area in the reference	SET-1 (pixels)	SET-2 (pixels)
Water	2	30,000	2,400
Vegetation	8	30,000	9,600
Built-up	45	30,000	54,000
Soil	45	30,000	54,000

4.2. Classifier Level Approach

The classifiers considered here can be classified as either pixel-based or patch-based. All the classifiers here assume a within-class balance. If a within-class imbalance is assumed, then a greater number of pixels are required for training, which may not be possible with multispectral imagery. The pixel-based classifiers are first implemented in a parametric way, where the information required to train the classifier is extracted from the training data. There are no hyperparameters in the parametric classifiers, no amount of increase in the training data can change the accuracy of the classifier. On the other hand, the hyperparameters of the non-parametric classifiers can be tuned to achieve better accuracy.

4.2.1. Pixel-Based Parametric Maximum Likelihood Classifier

The ML classifier assumes a Gaussian distribution; hence, each class is characterized by a mean vector and a covariance matrix. The classifier calculates the discriminating function depending on the *covariance matrix*, which is calculated from the information extracted from the training. This discriminating function is a statistical probability of the pixel belonging to each class. The pixel is assigned to a class that has the highest probability. Though the classifier is easy to implement, the assumption of Gaussian distribution may not always hold true. Hence, the classification accuracy suffers. Also, when the classes are not highly separable (homogenous geographic area), the covariance matrix becomes unstable, which again degrades the performance of the classifier.

4.2.2. Pixel-Based Non-Parametric Multiclass Support Vector Machines

By default, Support Vector Machines are used for Binary Classification (Two class Problems). For multiclass classification, the problem is broken down into multiple binary classification cases, which are called one-vs-one. The number of classifiers required for a one-vs-one multiclass classification is given by the following formula.

$$\text{Number of Classifiers required} = \frac{n*(n-1)}{2} \tag{1}$$

In the present context, we have 4 land cover classes; hence, we need 6 classifiers. The kernel function used can be linear, polynomial or Radial Basis Functions. The kernel function calculates the distance between two sets of data points (Two class problems). This distance is used to map the data points to a higher dimension for easier separability.

Parameter Tuning in SVM

In a pixel-based nan-parametric SVM classifier, the parameters can be tuned to achieve better accuracy. Parameter tuning in SVM is a tedious process and is taken from trial and error in this work. The change in the accuracy of the classifier with different kernel functions used is shown in Figure 3a.

In the present context, the polynomial kernel gives a better accuracy keeping the other parameters constant. Figure 3b shows the variation in the accuracy with respect to the cost function nu (ν) for all three kernels. The maximum accuracy is achieved with a value of ν= 0.7. Hence, the parameters chosen for training are kernel = polynomial and ν= 0.7.

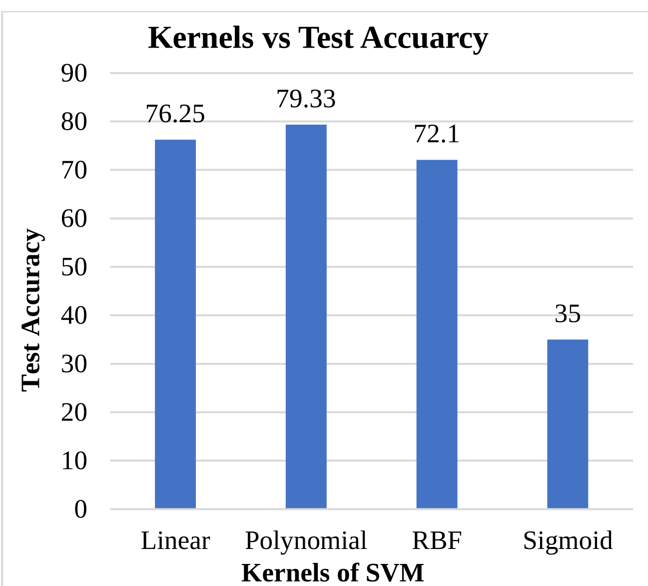

Fig. 3a. Kernels of SVM vs Test accuracy

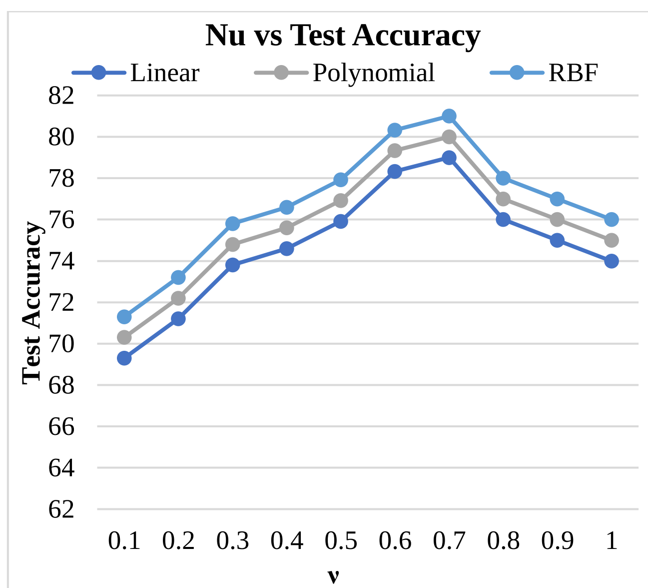

Fig. 3b. nu(ν) vs Test accuracy for SVM

4.2.3. Patch-based Convolutional Neural Networks

A convolution neural network is a deep learning architecture which takes an input image and assigns weights to the various aspects of the object under consideration in the image. In short, CNN learns the features of the images on the fly. In a patch-based CNN, there is a single image under consideration (a combined raw satellite image in this case). The image is then clipped to small squares, such that each square represents any one particular land cover class. The image clips around the geographical border will not be of the same size. There is a trade-off between the size of the clips and the processing time. The bigger the size of the clip, the lesser will be the processing time.

But in the case of an unbalanced class dataset, a bigger clip will not be able to represent a particular land cover class in its entirety or singly. For example, in this work, choosing a patch size of 128 x 128 results in a patch where, along with the class water, the class built-up is also included. This is because the class water is in the minority compared to the class built-up.

Hence, a patch of size 64 x 64, which represents only one class in the entire patch, is chosen. The detailed implementation of the proposed architecture is shown in Figure 4. A sequential model consisting of fourteen layers is proposed for the land cover classification. With this type of modelling, it is possible to build the model layer by layer. We use the following layers to build our network. The network was implemented with a keras framework with a tensorflow backend.

The entire network is divided into two parts after the input layer. The first part deals with the feature extraction from the patches. The convolution layer, the Dropout layer, the Global pooling layer and the Flatten layer are responsible for feature extraction. The second part is for the classification depending on the features extracted. This part has the Dense layer of the Fully connected layer. Here, the Relu Activation for all the layers except the dense Layer, which uses the softmax activation.

Input layer

As described previously, a patch size of 64 x 64 is considered for a 3-band combination of 4-3-2. This band combination represents the true colour composite.

Convolution Layer

Four convolution layers separated by the dropout layer are used in this work. Each convolution layer has a kernel size of 7 x 7. The first convolution layer (Conv1) has 32 such filters. This layer is hence represented as 7 x 7 x 32. The second layer (Conv2) has 48 filters. The third layer (Conv3) has 64 filters, and the fourth layer has 128 filters.

With each convolutional layer the spatial dimensions are reduced while increasing the depth of the feature maps. There is an increase in the number of filters used in the subsequent stages as the patterns become more and more complex. In order to capture the combination of all these patterns, the number of filters needs to be increased to create a kind of convolution pyramid.

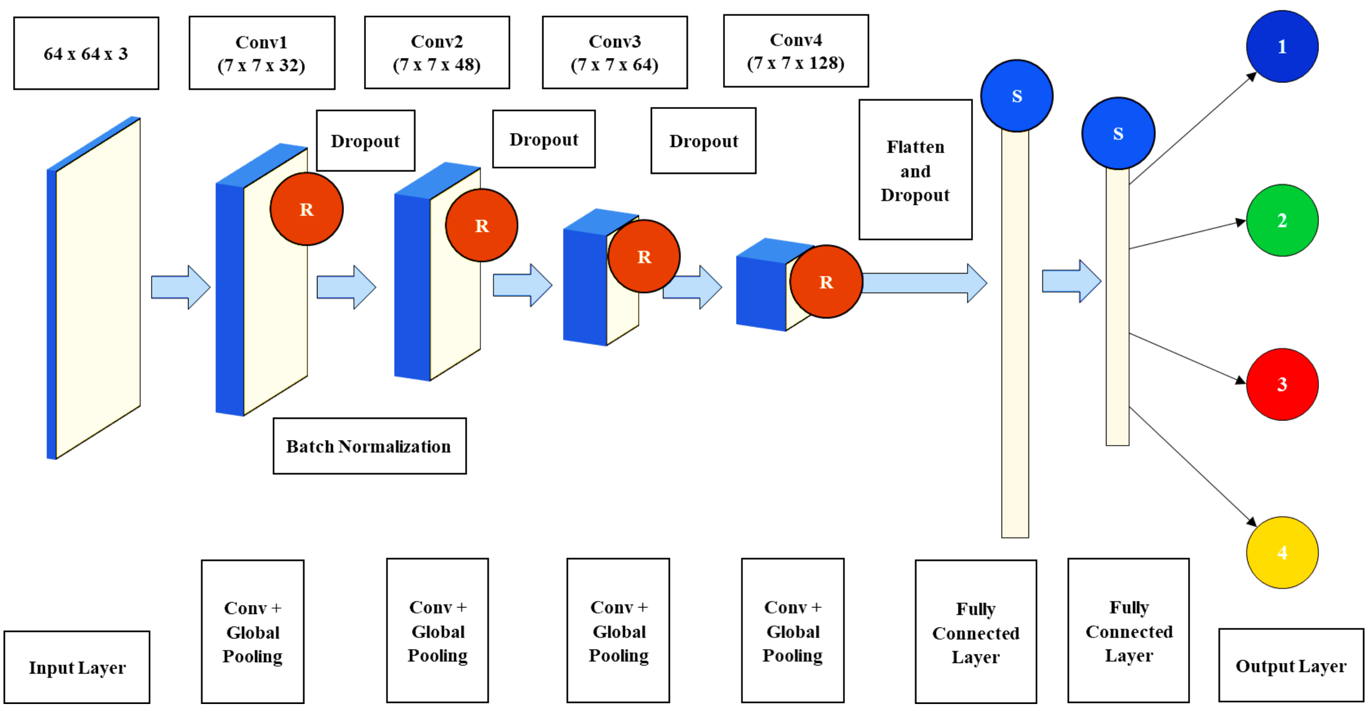

Fig. 4 Architecture of the proposed Convolutional Neural Network

Batch Normalization Layer

It ensures a stable distribution of the activation values throughout the training. It also speeds up the learning process even at higher learning rates. Batch Normalization is done before activation to reduce the runtime.

Global Pooling Layer

Generally used immediately after the activation function has been applied to the convolution layer. The pooling layer operates upon each feature map separately to create a new set of pooled feature maps. Global pooling down samples the entire feature map to a single value. This kind of functionality is more suited to the land cover classification. It can be used to aggressively summarize the presence of a feature in an image. It is sometimes used as an alternative to a fully connected network to transition from feature maps to output prediction for the model. Pooling reduces the size of the feature map. Here, it is performed with a 2 x 2 window, stride 2, and no padding.

Dropout Layer

Dropout is a regularization technique to prevent overfitting. During the training time, at each iteration, a neuron is temporarily dropped or disabled with a probability of ‘p’. The value of p chosen here is 0.25.

Softmax Layer

Generally, a sigmoid function is used for binary classification. In this work, we deal with multiclass classification; hence, a softmax layer is used.

Flatten Layer

Convert the 2D features to a 1D array so that it can be further processed by the dense layer.

Dense Layer

It is also called the Fully Connected Layer. The model is compiled with a sparse categorical cross entropy loss function and rms prop optimizer. The metric chosen is accuracy. The accuracy is displayed in the form of a confusion matrix, from which the overall accuracy, and user and Producer accuracy can be extracted. The model is trained with a batch size of 1 (Stochastic Gradient Descent) and 10 epochs.

5. Results

Figure 5a shows the LULC map obtained using the pixel-based ML classifier. Figure 5b shows the LULC map obtained using the SVM classifier, and Figure 5c shows the LULC map obtained using the CNN network. To prove the validity of better performance of a patch-based CNN compared to other classifiers, a part of the LULC map has been zoomed in.

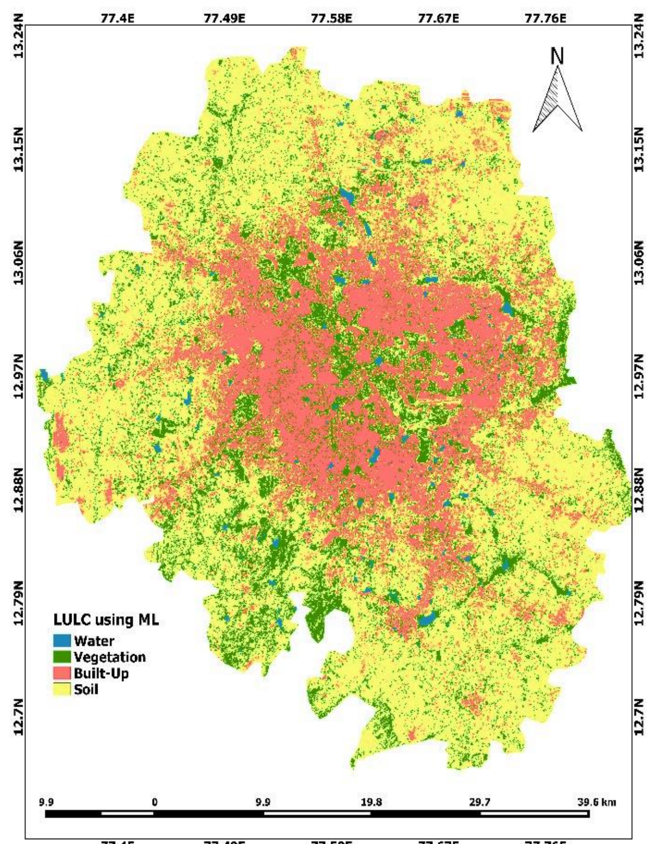

Fig. 5(a) LULC using ML classifier

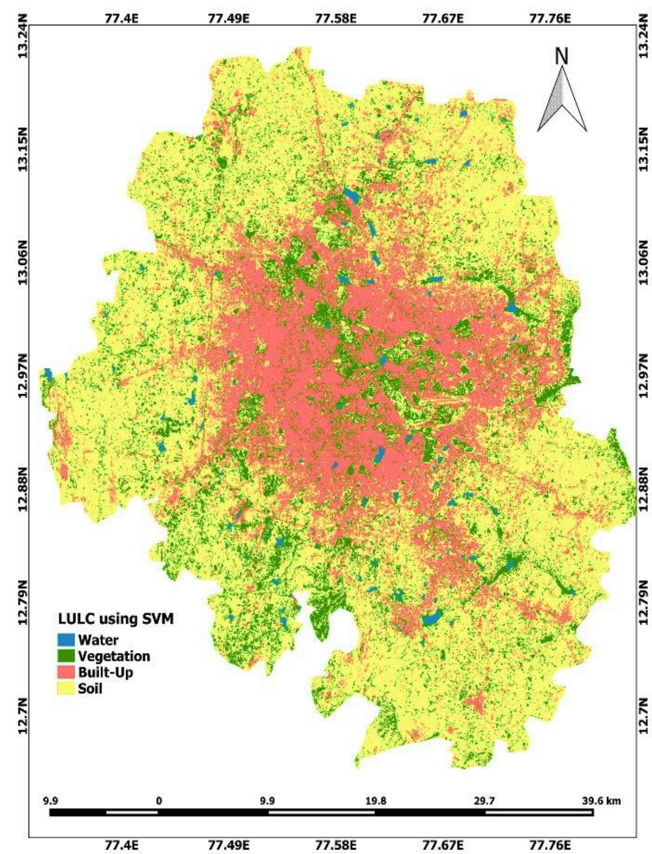

Fig. 5(b) LULC using SVM classifier

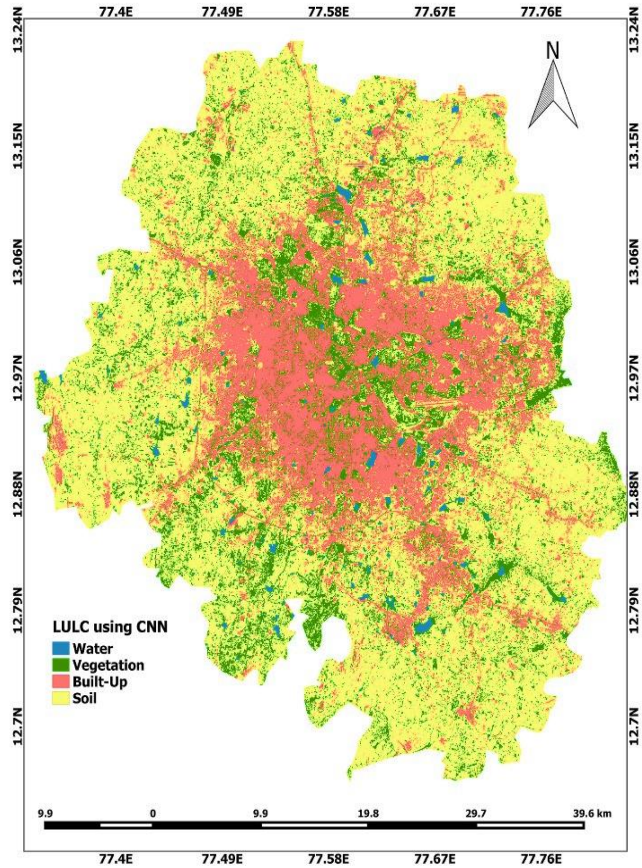

Fig. 5(c) LULC using CNN

Figure 6 shows the zoomed part of the raster in the band combination 4-3-2 with respect to the classes and the classification/misclassification of the three classifiers under consideration. The first row in Figure 6 indicates a part of the raster and the LULC map of the three classifiers, where a curved road is dominant. The second row indicates a part of the raster and the LULC map obtained with the three classifiers where the built-up area is dense. Similarly, the third row indicates the classification/misclassification between the classes water and vegetation. The overall accuracy of the ML classifier was 75.81% for area-proportional training data, whereas the accuracy was less (73.21%) when uniformly sampled training data was used. This proves that the area-proportional approach is better in terms of accuracy (Colditz, 2015), and hence, this method of sampling was used to train the SVM which gives an accuracy of 83%. Finally, CNN gives an overall accuracy of 91.2%.

6. Discussion

The main objective of this research work was to tackle the imbalanced classes problem using two different approaches and compare the approaches. To achieve this objective, a pixel-based parametric ML classifier was trained on area-proportional and uniformly sampled training data. It was found that the area-proportional sampling approach produces better results (75.81%). Though there is no scope to change the parameters and increase the accuracy of an ML classifier, hyperparameters can be tuned in SVM to increase the classification accuracy. A further scope in improving the classification accuracy is implementing a sequential patch-based CNN. There are several parameters which can be played around with to achieve a better classification accuracy.

The results in Figure 6 indicate that both SVM and CNN are successful in mapping the curved road (*C* & *D* in Figure 6), whereas ML classifiers fail here. The density of the built-up areas is mapped correctly by the ML classifier and CNN, whereas SVM fails here (second row). From the second row, it can also be seen that there are built-up areas around the curved road. ML classifies the density of the built-up correctly but fails to classify the curved road because the ML classifiers fail to take into consideration the neighbours of pixels. SVM does vice-versa. Only CNN classifies both things correctly. The third row of Figure 6 indicates the misclassification between the water and vegetation classes.

The water bodies are classified as vegetation by the ML and SVM classifier, whereas CNN classifies the water bodies correctly. The results obtained here also agree with Lee et al., 2016 and Amini et al., 2022 who showed that the classifier-based approach produces better results than the data-based approach.[1] This result can also be compared with the results obtained on a similar dataset (Kulkarni and Vijaya, 2021), where the RF classifier gives an accuracy of 87.13%. This is because the CNN is patch-based, where the model training is based on the patch image under consideration. Table 2 shows a detailed comparison of this work with other works available in the literature.

Table 2. Comparison of the results with other works in literature

Sl. No	Classifiers	Authors [Reference Numbers]	Methodology Used	Accuracy Obtained (%)	Accuracy (in this Research work) %
1	Support Vector Machine	Pradhan B [27]	SVM implemented for landslide susceptibility mapping	82.04	91.2
2	Fusion Classifier	Sitthi A., et al. [31]	Naïve Bayes Classifier integrated with external descriptors	87.94	

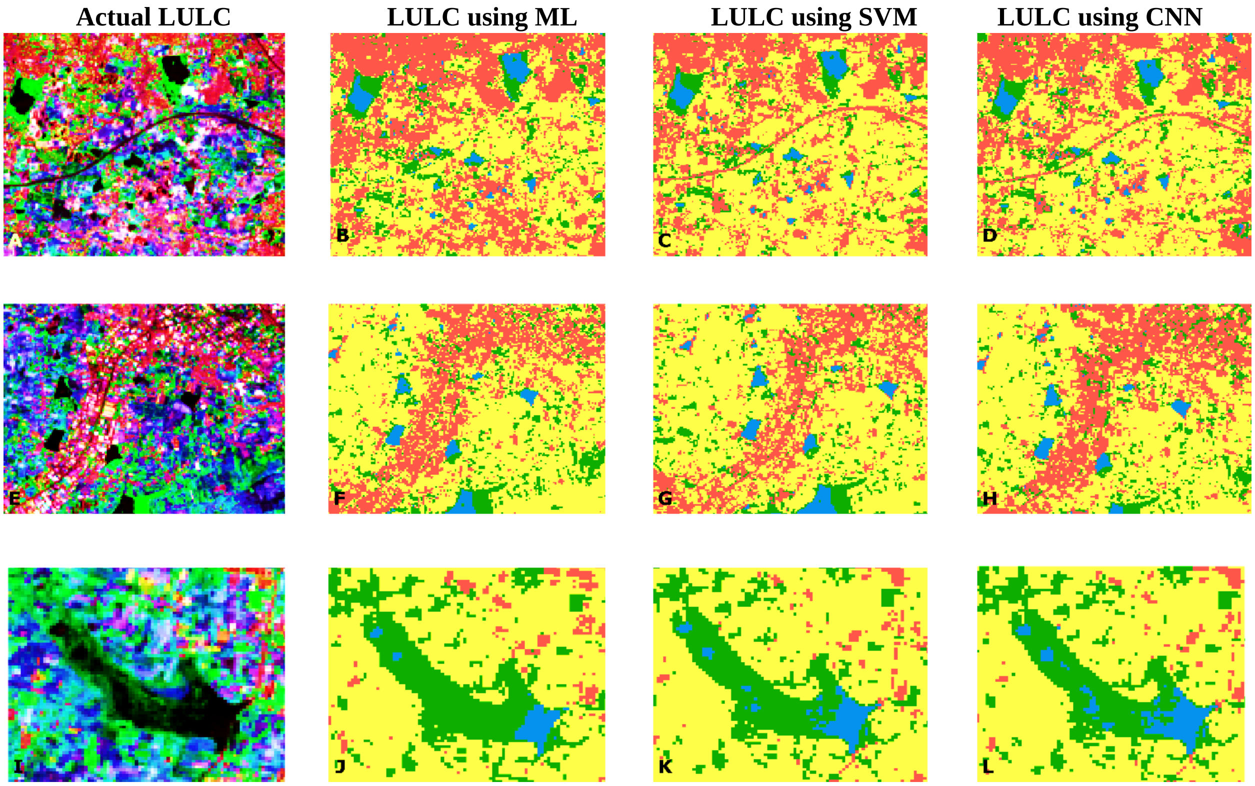

Fig. 6 Comparison of the methods with respect to classification/misclassification

7. Conclusion and Future Work

This work tackles the imbalanced classes problem on two levels, namely, the data level and the classifier level. It is shown that the classifier-level approach (CNN) is far more effective than the data-level approach (ML and SVM). The pixel-based parametric classifiers like ML perform poorly (accuracy = 75.81%) compared to the patch-based CNN for the LULC classification, where the dataset contains imbalanced classes.

Starting with the ML classifier, where there is no scope to play around with the parameters, the work proceeds to show that the parametric pixel-based classifiers, like SVM, perform better (accuracy = 83%) because of hyperparameter tuning. Patch-based CNN provides the highest accuracy of 91.2%. As a future work, the patch-based CNN can incorporate the data level approach by using data augmentation, which may further increase the classification accuracy. Application of evaluation methods which do justice to the imbalanced class classifiers can also be taken up as a future work.

Acknowledgements

The authors thank the management of BNM Institute of Technology, Bangalore, for providing the necessary resources and the infrastructure needed to carry out the research work. The authors are also thankful to Visveswaraya Technological University for providing a platform to carry out the research work.

References

[1] Saeid Amini et al., "Urban Land Use and Land Cover Change Analysis Using Random Forest Classification of Landsat Time Series," *Remote Sensing*, vol. 14, no. 11, pp. 1-23, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Akanksha Balha et al., "A Comparative Analysis of Different Pixel and Object-Based Classification Algorithms Using Multi-Source High Spatial Resolution Satellite Data for Lulc Mapping," *Earth Science Informatics*, vol. 14, pp. 2231-2247, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3] Paula D. Blanco et al., "A Land Cover Map of Latin America and the Caribbean in the Framework of the Serena Project," *Remote Sensing of Environment*, vol. 132, pp. 13-31, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[4] Rok Blagus, and Lara Lusa, "Class Prediction for High-Dimensional Class-Imbalanced Data," *BMC Bioinformatics*, vol. 11, pp. 1-17, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[5] Christina Bogner et al., "Classification of Rare Land Cover Types: Distinguishing Annual and Perennial Crops In an Agricultural Catchment in South Korea," *PLoS One*, vol. 13, no. 1, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] N.V. Chawla et al., “SMOTE: Synthetic Minority Over-sampling Technique,” *Journal of Artificial Intelligence Research*, vol. 16, pp. 321-357, 2002. [CrossRef] [Google Scholar] [Publisher Link]

[7] René Roland Colditz, “An Evaluation Of Different Training Sample Allocation Schemes for Discrete and Continuous Land Cover Classification using Decision Tree-Based Algorithms,” *Remote Sensing*, vol. 7, no. 8, pp. 9655-9681, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[8] Georgios Douzas, and Fernando Bacao, “Geometric SMOTE a Geometrically Enhanced Drop-in Replacement for SMOTE,” *Information Sciences*, vol. 501, pp. 118-135, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] Martina Deur, Mateo Gašparović, and Ivan Balenović, “Tree Species Classification in Mixed Deciduous Forests Using Very High Spatial Resolution Satellite Imagery and Machine Learning Methods,” *Remote Sensing*, vol. l2, no. 23, pp. 1-18, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[10] Wei Feng, Wenjiang Huang, and Wenxing Bao, “Imbalanced Hyperspectral Image Classification with an Adaptive Ensemble Method Based on SMOTE and Rotation Forest with Differentiated Sampling Rates,” *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 12, pp. 1879-1883, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] Alberto Fernández et al., “Analysing the Classification of Imbalanced Data-Sets with Multiple Classes: Binarization Techniques and Ad-Hoc Approaches,” *Knowledge-Based Systems*, vol. 42, pp. 97-110, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[12] Joao Fonseca, Georgios Douzas, and Fernando Bacao, “Improving Imbalanced Land Cover Classification with K-Means SMOTE: Detecting and Oversampling Distinctive Minority Spectral Signatures,” *Information*, vol. 12, no. 7, pp. 1-20, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Kpade O.L. Hounkpatin et al., “Predicting Reference Soil Groups Using Legacy Data: A Data Pruning and Random Forest Approach for Tropical Environment (Dano Catchment, Burkina Faso), *Scientific Reports*, vol. 8, pp. 1-16, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[14] Huiran Jin, Stephen V. Stehman, and Giorgos Mountrakis, “Assessing the Impact of Training Sample Extraction on Accuracy of an Urban Classification: A Case Study in Denver, Colorado,” *International Journal of Remote Sensing*, vol. 35, no. 6, pp. 2067-2081, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[15] Taeho Jo, and Nathalie Japkowicz, “Class Imbalances versus Small Disjuncts,” *ACM Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 40-49, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[16] Shahab Eddin Jozdani, Brian Alan Johnson, and Dongmei Chen, “Comparing Deep Neural Networks, Ensemble Classifiers, and Support Vector Machine Algorithms for Object-Based Urban Land Use/Land Cover Classification,” *Remote Sensing*, vol. 11, no. 14, pp. 1-24, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Harsurinder Kaur, Husanbir Singh Pannu, and Avleen Kaur Malhi, “A Systematic Review on Imbalanced Data Challenges in Machine Learning: Applications and Solutions,” *ACM Computer Surveys*, vol. 52, no. 4, pp. 1-36, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[18] Land Use / Land Cover Analysis-Third Cycle Technical, Natural Resource Census, pp. 1-12, 2019. [Online]. Available: https://bhuvan-app1.nrsc.gov.in/2dresources/thematic/LULC503/lulc.pdf

[19] Taehyung Lee, Ki Bum Lee, and Chang Ouk Kim, “Performance of Machine Learning Algorithms for Class-Imbalanced Process Fault Detection Problems,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 29, no. 4, pp. 436-445, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[20] Congcong Li et al., “Comparison of Classification Algorithms and Training Sample Sizes in Urban Land Classification with Landsat Thematic Mapper Imagery,” *Remote Sensing*, vol. 6, no. 2, pp. 964-983, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[21] Julián Luengo et al., *Imbalanced Data Preprocessing for Big Data*, Big Data Preprocessing, Springer, Cham, pp. 147-160, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[22] Aaron E. Maxwell, Timothy A. Warner, and Fang Fang, “Implementation of Machine-Learning Classification in Remote Sensing: An Applied Review,” *International Journal of Remote Sensing*, vol. 39, no. 9, pp. 2784-2817, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[23] Varun Narayan Mishra et al., “Performance Evaluation of Textural Features in Improving Land Use/Land Cover Classification Accuracy of Heterogeneous Landscape Using Multi-Sensor Remote Sensing Data,” *Earth Science Informatics*, vol. 12, pp. 71-86, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[24] Ranjit Panigrahi et al., “A Consolidated Decision Tree-Based Intrusion Detection System for Binary and Multiclass Imbalanced Datasets,” *Mathematics*, vol. 9, no. 7, pp. 1-35, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Fernando Pech-May et al., “Mapping of Land Cover with Optical Images, Supervised Algorithms, and Google Earth Engine,” *Sensors*, vol. 22, no. 13, pp. 1-19, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[26] Kwanele Phinzi et al., “Understanding the Role of Training Sample Size in the Uncertainty of High-Resolution Lulc Mapping Using Random Forest,” *Earth Science Informatics*, vol. 16, pp. 3667-3677, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[27] Biswajeet Pradhan, "A Comparative Study On The Predictive Ability of the Decision Tree, Support Vector Machine and Neuro-Fuzzy Models in Landslide Susceptibility Mapping Using GIS," *Computers & Geosciences*, vol. 51, pp. 350-365, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[28] Julien Radoux et al., "Automated Training Sample Extraction for Global Land Cover Mapping," *Remote Sensing*, vol. 6, no. 5, pp. 3965-3987, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[29] Amin Sharififar, Fereydoon Sarmadian, and Budiman Minasny, "Mapping Imbalanced Soil Classes Using Markov Chain Random Fields Models Treated With Data Resampling Technique," *Computers and Electronics in Agriculture*, vol. 159, pp. 110-118, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[30] Yuan-Hai Shao et al., "An Efficient Weighted Lagrangian Twin Support Vector Machine for Imbalanced Data Classification," *Pattern Recognition*, vol. 47, no. 9, pp. 3158-3167, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[31] Asamaporn Sitthi et al., "Exploring Land Use and Land Cover of Geotagged Social-Sensing Images Using Naive Bayes Classifier," *Sustainability*, vol. 8, no. 9, pp. 1-22, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[32] M. Zhu et al., "Land Cover Classification Using High Resolution Satellite Image Based on Deep Learning," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 42, pp. 685-690, 2020. [CrossRef] [Google Scholar] [Publisher Link]