Spatial-temporal analysis of urban land use, land cover using neural network model across Indian cities and time periods

Anand Prakash, Anindita Das Gupta, Prof. Uttam Kumar

Abstract

Urban land cover classification plays a crucial role in urban planning, environmental monitoring, and land management. In this study, we present a comprehensive analysis of urban land cover classification using multispectral satellite imagery and neural network modelling. The focus of this research is on training a neural network model utilising both Hyderabad 2021 and Kolkata 2021 datasets to classify urban land cover categories, including both four-class (built up, vegetation, barren and water) and two-class (built up and non-built up) classifications. Leveraging the spectral information from multispectral bands, a dense neural network architecture is employed to learn the complex relationships between input features and land cover labels. Subsequently, we apply the trained model to predict land cover classifications for Bangalore in the years 2013, 2017, and 2021.

Keywords: land use land cover classification, temporal analysis, remote sensing, statistical analysis, feed-forward neural network, cross-validation

Introduction

In the last few decades, satellites have been equipped with hundreds of sensors to capture large amounts of data with high spatial, temporal and spectral resolutions. Remote Sensing (RS) imagery is the most practical and economical method of predicting land use and land cover (LULC) maps [1] [2]. The prediction of LULC previously relied on ground truth creation was based on field observations. Creation of large training samples for supervised classification is laborious and time consuming [3]. Additionally extraction of ground truth from the satellite images requires significant human, material resources and knowledge. When compared to traditional field surveys, the determination of physical characteristics of the earth's surface through the analysis of images has resulted in a significant reduction in both time and expense. To mitigate the effects of land use and land cover changes, it is critical to more precisely detect LULC changes [4]. Techniques that are more reliable and simple to use must be devised in order to quantify such changes [5].

Many algorithms have been developed to analyse RS data and extract meaningful information and insights from the data that describe changes and developments in land covers [4][5]. The analysis of RS images is challenging due to its high‑dimensionality and large volumes of these images. Classifying LULC is a key function of different algorithms used in RS. The automated extraction of data from the earth's surface is crucial for several RS applications including LULC change detection, environmental monitoring, urban expansion, surveillance of crops, coastal changes, flood risk assessment etc [3].

The task of giving each pixel in an image a class label is known as training dataset creation which is crucial in LULC classification. For time series LULC supervised classification the first task is to gather training datasets for each class from different representative samples and each year separately which can be tedious. Due to the diverse look and high intra-class variance of the elements, the abundance of spectral channels and the small number of labelled samples in the images classification frequently turns out to be a difficult process [4].Remote sensing image classification, anomaly detection and prediction issues can be resolved with machine learning (ML) modelling. A variety of machine learning (ML) techniques, such as support vector machines (SVM), Markov chain models, k-nearest neighbour, Maximum Likelihood, Random Forest classifiers etc. are used to classify images [6]. Majority of research has utilised the variants of Artificial Neural Networks (ANN), Markov Chain and statistical models to simulate and predict the Land Use and Land Cover (LULC) changes from RS data. However most existing research works have some limitations in terms of accuracy and the need for intervention of users in the training process. A novel modelling technique that can manage enormous amounts of data and better anticipate spatial-temporal characteristics through deep learning (DL) has been introduced as a result of the growth of ML modelling techniques and the amount of earth data [7]. The emergence of deep learning (DL) and its ability to learn representations of data and spatial features automatically from multilayer networks has provided good results in many fields such as image recognition, classification, segmentation and prediction[8] [9]. DL model has outperformed traditional models when it comes to extracting spatial multilevel features from remote sensing images [10].

In this work, different years and distinct regions image classification was done using the Convolution Neural Networks (CNN) model. Landsat 8 OLI/TIRS Collection 2 atmospherically corrected surface reflectance, tier 1 dataset was used for the analysis. Random Forest (RF) supervised classification algorithm was utilised to classify Hyderabad and Kolkata into four classes: vegetation, water, build-up, and other classes. For this purpose an equal number of training sets for all the classes were created in Google Earth Engine (GEE). Random forest is an ensemble learning-based supervised machine learning algorithm. Instead of using individual models to create predictions, ensembles use a collection of models. The random forest ensemble model, which combines a huge number of decision trees, is arguably the most prominent of the ensemble models. Ensemble learning is a method of learning in which multiple versions of the same algorithm are combined to form a more efficient prediction model.

Numerous study avenues have been made possible by the vast diversity of datasets that are available which has also sparked the creation of novel methods for digital pattern classification. The study's contribution is to predict LULC based on different sources of data from satellite images where training the model is not necessary every time. We intend to investigate the possibilities of streamlining the training phase by leveraging the classified image of a certain city or time period as we aim to understand the dynamics of city expansion.

Data collection methods

In this section, we will outline the process of data collection and preprocessing for our urban land cover classification study. We will begin by presenting the satellite images selected for analysis and discuss the necessary steps performed on the data before feeding it into the machine learning models. Furthermore, we will provide details on the sources of the data, including relevant descriptions, and share corresponding Google Earth Engine links for reference.

Description of datasets and their sources

We are using publicly available datasets for our study
The dataset used for this study is the USGS Landsat 8 Level 2, Collection 2, Tier 1 dataset. This dataset contains atmospherically corrected surface reflectance and land surface temperature derived from the data produced by the Landsat 8 OLI/TIRS sensors.

The Landsat 8 Level 2 dataset consists of the following key components:
1.Surface Reflectance (SR):
- This dataset includes 5 visible and near-infrared (VNIR) bands (SR_B2 to SR_B5) and 2 short-wave infrared (SWIR) bands (SR_B6 to SR_B7).
- These bands are processed to orthorectified surface reflectance, providing information about the amount of solar radiation reflected by the Earth's surface in different spectral bands.

2. Surface Temperature (ST):
- The dataset also contains one thermal infrared (TIR) band (ST_B10), which is processed to orthorectified surface temperature.
- This band provides information about the temperature of the Earth's surface in the thermal infrared region.

Multispectral Bands Used:
The multispectral bands utilised for feature extraction in this study are as follows:
- SR_B2: Band 2 (Blue) Surface Reflectance
- SR_B3: Band 3 (Green) Surface Reflectance
- SR_B4: Band 4 (Red) Surface Reflectance
- SR_B5: Band 5 (Near Infrared) Surface Reflectance

- SR_B6: Band 6 (Shortwave Infrared 1) Surface Reflectance
- SR_B7: Band 7 (Shortwave Infrared 2) Surface Reflectance
- ST_B10: Band 10 Surface Temperature

Band Descriptions:
- SR_B2 to SR_B7:These bands represent surface reflectance in various spectral regions, ranging from blue to shortwave infrared.
- ST_B10: This band provides surface temperature information in the thermal infrared region. If the processing level is set to 'L2SR', this band is fully masked out.

The dataset extraction for this study involves acquiring satellite images for specific time frames and geographical boundaries to facilitate urban land cover classification. The following details outline the time frames and geographical boundaries for data extraction:

1. Bangalore:
   - Satellite images for the years 2021, 2017, and 2013 are extracted for the city of Bangalore.
   - These images serve as the testing dataset for evaluating the performance of the trained models.

2. Hyderabad and Kolkata:
   - Satellite images for the year 2021 are extracted for the city of Hyderabad and Kolkata.
   - These images are utilized as the primary training dataset for model development.

Data generation

This data generation process encompasses image preprocessing, sampling.It prepares the satellite imagery and corresponding labels for subsequent machine learning tasks.

1. Image Preprocessing:
   - The script defines a function named `prepSrL8()` for preprocessing Landsat 8 Level 2 imagery. This function includes:
     - Masking unwanted pixels such as fill, cloud, and cloud shadow.
     - Applying scaling factors to the spectral bands for reflectance and surface temperature.

2. Region of Interest (ROI) Definition:
   - The script defines a region of interest (ROI) polygon using the `ROI` feature collection.
   - This polygon is used to clip the satellite imagery to the specific geographical area of interest.

3. Image Collection and Filtering:
   - The script utilises the `ee.ImageCollection` function to retrieve Landsat 8 imagery for the desired time frame (e.g., '2021-01-01' to '2022-01-01').

- The imagery is filtered based on the specified date range and processed using the `prepSrL8()` function for atmospheric correction and preprocessing.

4. Median Compositing:
   - The `median()` function is applied to the image collection to create a median composite image.
   - This composite image represents the median values of spectral bands for the selected time frame, providing a clear representation of the land cover.

5. Data Export:
   - The classified image and spectral bands of interest are exported to Google Drive as GeoTIFF files for further analysis and visualisation.

Data preparation

1. Flattening and Merging:
   - The spectral bands extracted from the satellite images are flattened to convert them into one-dimensional arrays.
   - The classified labels are also flattened to create one-dimensional arrays.
   - These flattened spectral bands and labels are then merged into a single array, forming the feature space for training the model.

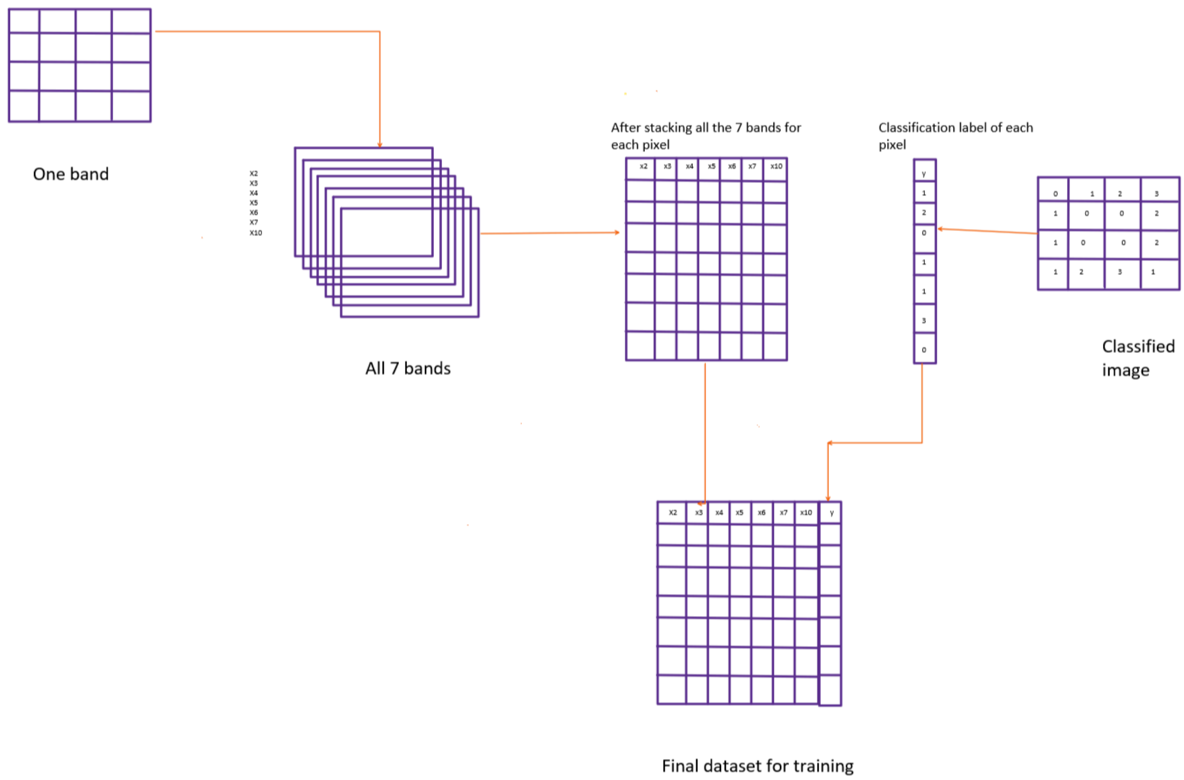
2. Handling Missing Values:
   - A check is performed to identify and count the number of NaN (Not a Number) values in the merged array.
   - Any NaN values present in the merged array are replaced with suitable values or handled appropriately before training the model.

3. Scaling:
   - The features (spectral bands) are scaled using the `StandardScaler` from scikit-learn.
   - Scaling ensures that all features have a similar scale, preventing certain features from dominating the model training process.

4. Train-Test Split:
   - The dataset is split into training and testing sets using the `train_test_split` function from scikit-learn.
   - This split allows for model training on one portion of the dataset and evaluation on another portion, enabling assessment of model performance.

Models information

Feature space formation



Stacking Multispectral Bands and Classification Labels for Pixel-Level Analysis:

In the context of multispectral satellite imagery analysis, each pixel within an image corresponds to a unique geographical location and is characterised by its spectral response across multiple spectral bands. These spectral bands capture electromagnetic radiation reflected or emitted from the Earth's surface, providing valuable information about the underlying land cover and land use characteristics.

Feature Space Construction

For each pixel within the satellite imagery of a city, there exist multiple spectral bands, typically ranging from visible to infrared wavelengths. In our dataset, this comprises a total of seven multispectral bands. These bands represent the radiometric intensity values across different wavelengths and serve as the primary features for analysis.

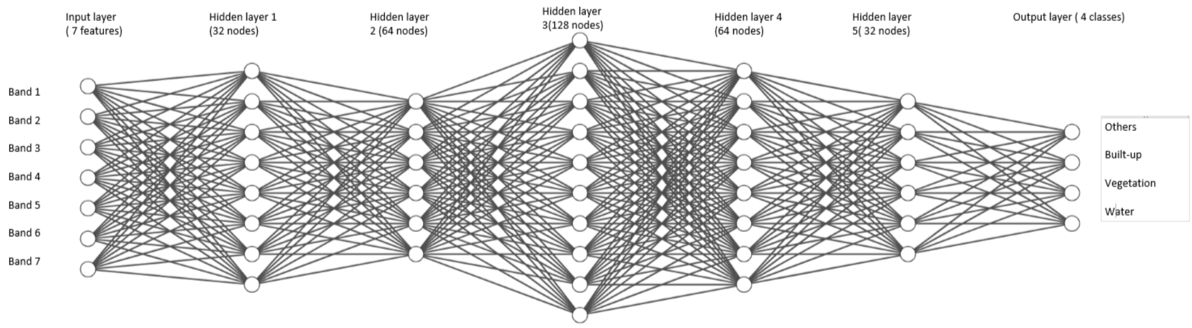
Classification Label Assignment:
Alongside the multispectral bands, each pixel is associated with a classification label that signifies its land cover or land use category. This classification label denotes whether the pixel belongs to categories such as barren land, built-up area, vegetation cover, or water body.

These labels are derived from prior classification or segmentation processes based on ground truth data or remote sensing algorithms.

Creation of the Final Dataset

To prepare the data for model training and testing, the multispectral bands and corresponding classification labels are stacked together to form the final dataset. For each pixel within the satellite imagery, the spectral values across the seven bands are concatenated into a single feature vector, representing the feature space of that pixel. Concurrently, the associated classification label is appended to this feature vector, establishing a one-to-one correspondence between spectral features and classification labels.

Neural network architecture



Neural network architecture (feed forward neural network)

The model architecture is a Feedforward Neural Network, specifically a Multi-Layer Perceptron (MLP), commonly used for classification tasks. It consists of six layers, excluding the input layer. Each layer has a specific number of neurons, contributing to the network's ability to learn complex patterns in the data.

The hidden layers, comprising 64, 32, 128, 64, and 32 neurons respectively, are responsible for processing the input data through non-linear transformations. These layers employ the Rectified Linear Activation function (ReLU), which introduces non-linearity to the model, enabling it to learn and represent complex relationships in the data more effectively.

The output layer's architecture is crucial for determining the model's predictions. The number of neurons in the output layer matches the number of classes in the dataset, which is determined by the shape of the target variable, y.shape[1]. In this configuration, the output layer uses the Softmax activation function. Softmax converts the raw output scores into probabilities, providing a probability distribution over the classes and facilitating the classification of input samples into the respective classes.

During training, the model aims to minimise the Categorical Crossentropy loss function. This loss function quantifies the difference between the predicted probability distribution and the actual distribution of class labels in the training data. Minimising the loss function guides the model to make more accurate predictions and improve its performance on unseen data.

To optimise the model's parameters and update them during training, the Adam optimizer is employed. Adam is an adaptive learning rate optimization algorithm that adjusts the learning rate for each parameter based on estimates of the first and second moments of the gradients. This adaptive learning rate helps accelerate convergence and enhances the model's training efficiency.

Mathematical representation

Let X be the input to the model with dimensionality (n, d) , where n is the number of samples and d is the number of features, in our case d is 7 .

Let u be the ground truth labels with dimensionality (n, c), where c is the number of classes(which is in our case is 4)
Let W_i and b_i represent the weights of and biases of the i^{th} layer respectively.

The mathematical form of the hidden layers is given by :
H_i = **ReLU** $(XW_i + b_i)$ $, i = 1, 2, 3, 4, 5$

The mathematical form of the output layer is given by :

Output = Softmax $(H_5 w_6 + b_6)$
The loss function used for training is categorical cross entropy , given by

Loss = $- (1/n) \sum_{i=1}^{n} \sum_{j=1}^{c} y_{ij} \, log \, (Output_{ij})$

Results:

RF supervised classified images of Hyderabad and Kolkata in the year 2021 were obtained independently in order to classify Bangalore for the year 2021. After classifying Bangalore for the year 2021, the city was subsequently classified also for the year 2017 and 2013. In this study the classified image of both the cities Kolkata and Hyderabad were utilised separately, to classify Bangalore into four classes. The result showed the city of Hyderabad performed better than Kolkata for training Bangalore. While taking Hyderabad for classifying Bangalore into four classes, the built-up class was underestimated because of spectral mixing of reflectance with the other class (fallow land, barren land and soil). Built-up and other classes have similar reflectance because of which misclassifications took place between these two classes.

The attempt was made to avoid creation of any training data set for Bangalore separately and utilise the Hyderabad and Kolkata classified datasets to train using DL to simplify the classification process.

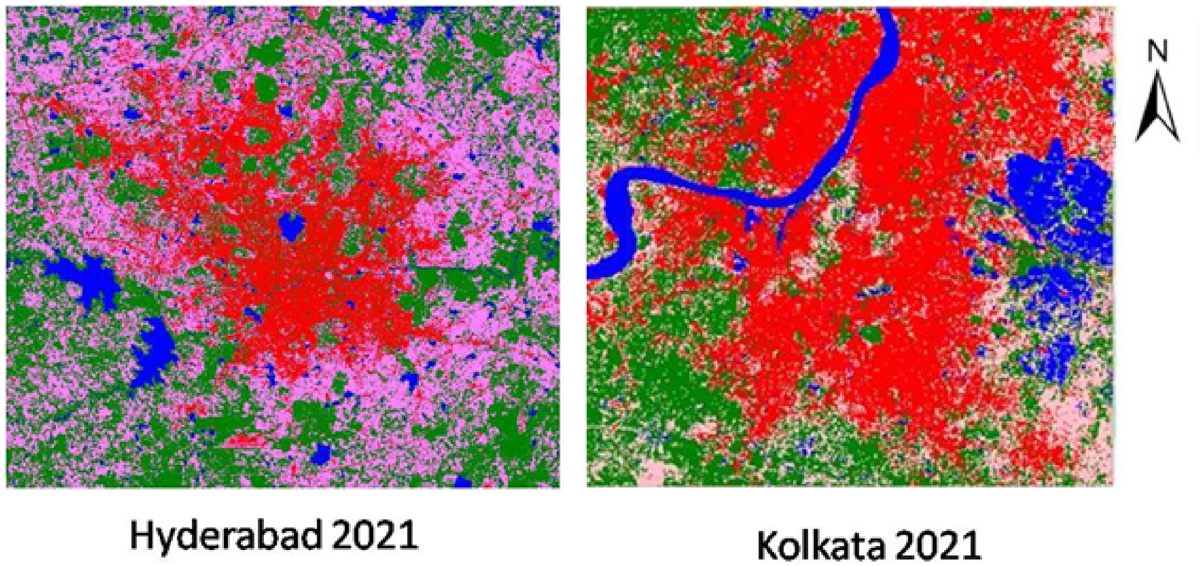
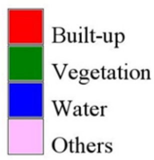
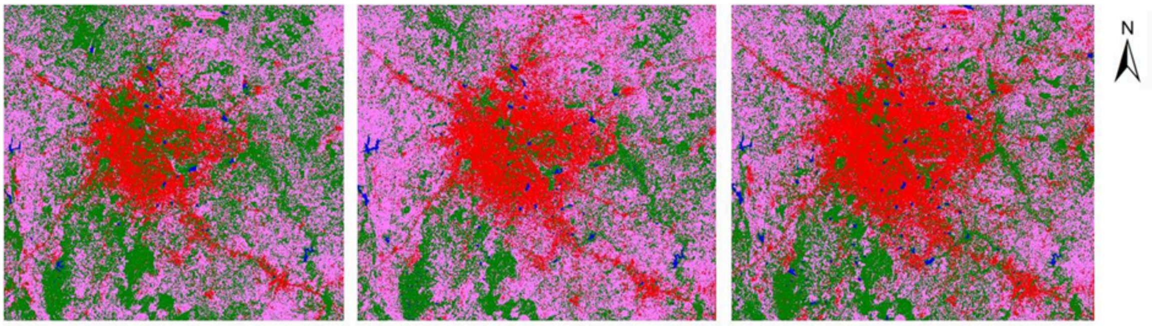


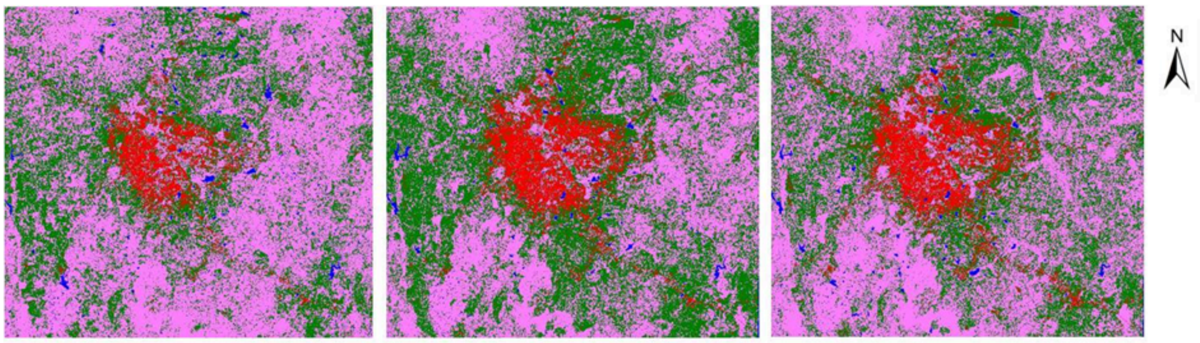
Fig1: Supervised classified image of Hyderabad and Kolkata





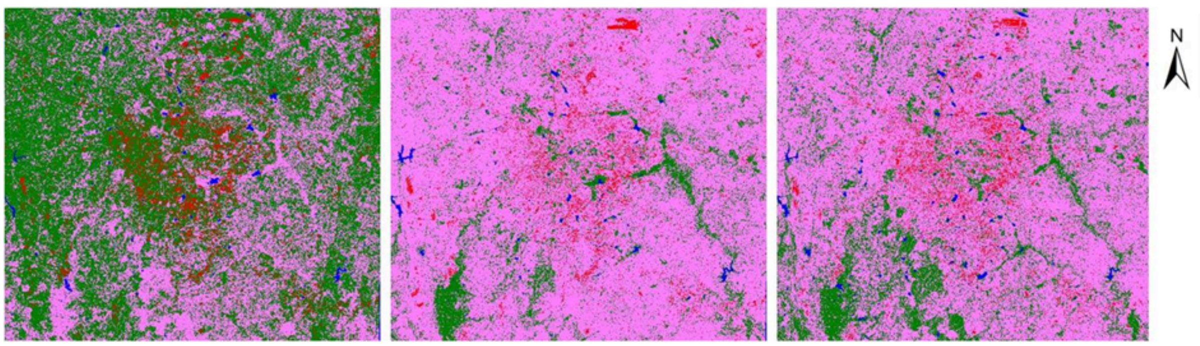
Bangalore 2013 Bangalore 2017 Bangalore 2021

Fig2: Supervised classified image of Bangalore



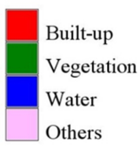
Predicted 2013 Predicted 2017 Predicted 2021

4 class classification (trained on Hyderabad)



Predicted 2013 Predicted 2017 Predicted 2021

4 class classification (trained on Kolkata)



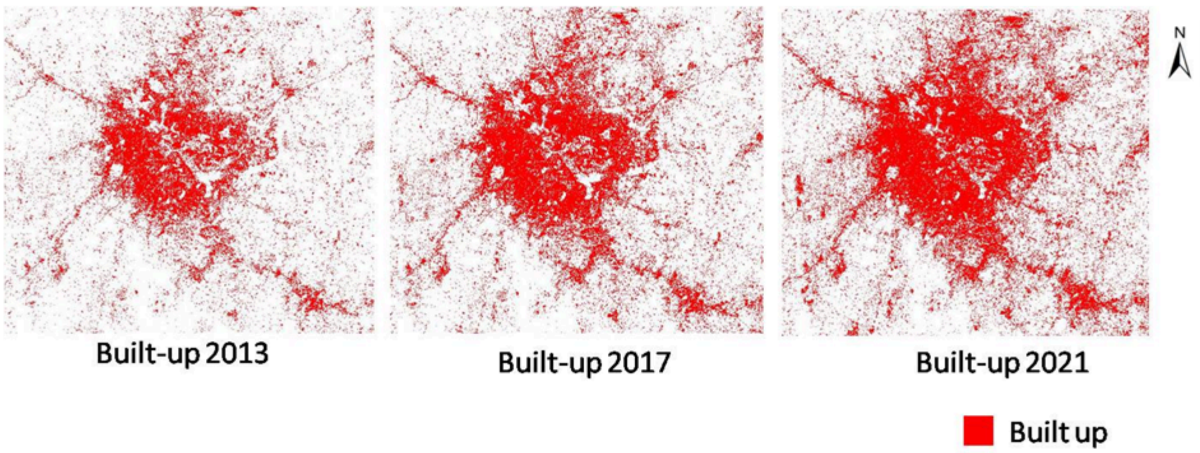
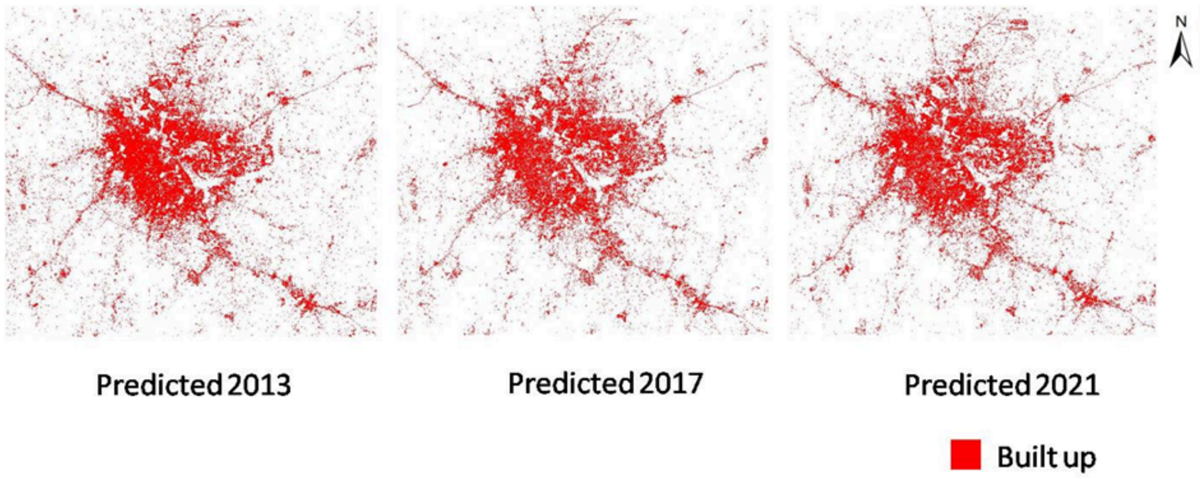
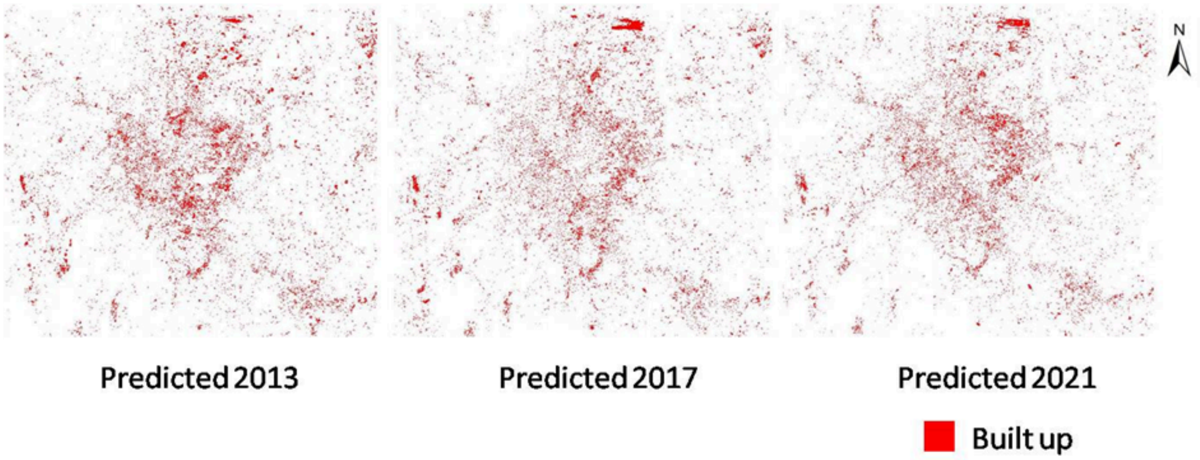


Fig5: Binary classified image Bangalore



Binary classification of Bangalore (Trained on Hyderabad)



Binary classification of Bangalore (Trained on Kolkata)

Conclusion

The data collection consists of multiple multi-date satellite images taken at different locations. In particular, we hope to offer insights, suggestions and decision support for future LULC production for urban growth through implementing, testing, and assessing a new approach. The suggested method will assist in determining changes in land cover and land use at a spatio-temporal scale.

References

[1] Hussain, M., Chen, D., Cheng, A., Wei, H., & Stanley, D. (2013). Change detection from remotely sensed images: From pixel-based to object-based approaches. ISPRS Journal of photogrammetry and remote sensing, 80, 91-106.

[2] He, C., Liu, Z., Tian, J., & Ma, Q. (2014). Urban expansion dynamics and natural habitat loss in China: A multiscale landscape perspective. Global change biology, 20(9), 2886-2902.

[3] Carranza-García, M., García-Gutiérrez, J., & Riquelme, J. C. (2019). A framework for evaluating land use and land cover classification using convolutional neural networks. Remote Sensing, 11(3), 274.

[4]Digra, M., Dhir, R., & Sharma, N. (2022). Land use land cover classification of remote sensing images based on the deep learning approaches: a statistical analysis and review. Arabian Journal of Geosciences, 15(10), 1003.

[5]Boulila, W., Ghandorh, H., Khan, M. A., Ahmed, F., & Ahmad, J. (2021). A novel CNN-LSTM-based approach to predict urban expansion. Ecological Informatics, 64, 101325.

[6]Aburas, M. M., Ahamad, M. S. S., & Omar, N. Q. (2019). Spatio-temporal simulation and prediction of land-use change using conventional and machine learning models: a review. Environmental monitoring and assessment, 191(4), 205.

[7] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015. https:// doi. org/ 10. 1038/ natur e14539

[8] Balarabe, A. T., & Jordanov, I. (2021, July). LULC image classification with convolutional neural network. In 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS (pp. 5985-5988). IEEE.

[9] C. Pelletier, G. I. Webb, and F. Petitjean, "Temporal convolutional neural network for the classification of satellite image time series," Remote Sensing, vol. 11, no. 5, pp. 523, 2019.

[10]Zhang, L., Zhang, L., & Du, B. (2016). Deep learning for remote sensing data: A technical tutorial on the state of the art. IEEE Geoscience and remote sensing magazine, 4(2), 22-40.