S & M 3453

Land Cover Mapping of Bengaluru's Urban and Surrounding Area with Spatiotemporal Open-source Remote Sensing Data

Sylvain Grison,[1] Rajath Siddaganga,[2,3] Renu Singh,[4] Shrihari Hegde,[2,3]
Axel Brockmann,[2] Smitha Krishnan,[3] and Wei Guo[1*]

[1]The University of Tokyo, 7 Chome-3-1 Hongo, Bunkyo City, Tokyo 113-8654, Japan
[2]National Centre for Biological Sciences - Tata Institute of Fundamental Research,
Bellary Road, Bengaluru 560065, Karnataka, India
[3]Bioversity International, College of Horticulture, UHS Campus, GKVK Post, 560065, Bengaluru, India
[4]Google, 8 Rue de Londres, 75009 Paris, France

(Received August 28, 2023; accepted November 20, 2023)

Keywords: land cover, Bengaluru, open data, remote sensing, urban area

The rapid urbanization observed in Asian tropics has resulted in extensive landscape transformations, giving rise to novel challenges such as conflicts of interest among citizens and threats to biodiversity. To facilitate informed urban management policies, there is a pressing need for contemporary land use and cover maps that provide precise insights into the evolving landscape. In this paper, we present the city of Bengaluru, India (covering an area of 365,879 ha) as a case study. We introduce an open-source-driven pipeline method capable of generating an updated 11-class land cover map at a high resolution of 10 m, achieving a global $F1$ score of 0.82. Notably, our proposed pipeline represents a pioneering solution that effectively addresses the persistent issue of cloud cover caused by monsoons, enhancing the utility of such maps for urban management, and planning in rapidly evolving regions.

1. Introduction

The rapid urbanization witnessed in Asian tropics has given rise to a unique set of challenges. Among these challenges, one noteworthy concern is the escalating human–bee conflict stemming from the encroachment of urban expansion into areas critical for ecosystem services and the survival of insect populations. In particular, colonies of the giant honey bee, *Apis dorsata*, have increasingly targeted surfaces within Indian urban centers like Bengaluru for nesting and foraging. *Apis dorsata* exhibits a distinct preference for nesting in elevated structures such as high-rise apartments and tall urban trees. However, owing to their inherent aggressiveness and substantial physical size (often twice that of an *Apis mellifera* individual), these bees evoke fear among the populace, prompting calls to pest management services for the destruction of their nests. Recognizing the vital role played by *Apis dorsata* as pollinators of plants and crops in the Bengaluru region,(1) it has become imperative to protect and conserve these important pollinators. To address these complex urban challenges within a megacity such

*Corresponding author: e-mail: guowei@g.ecc.u-tokyo.ac.jp

https://doi.org/10.18494/SAM4627

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

as Bengaluru, we have undertaken the task of creating a comprehensive land cover map to be able to study their precise nesting and foraging areas. Additionally, we aim to establish a transparent pipeline that enables the regular updating of this map, providing valuable resources for researchers and urban policymakers alike as they strive to mitigate the human–bee conflict and other associated issues.

The land use land cover (LULC) represents the land surface type, such as forests, water, built-up, or rocky areas. The increasing availability of high-temporal- and high-spectral-resolution data from artificial Earth observation (EO) satellites and the emergence of powerful algorithms and processing platforms to take advantage of a high volume of data have made remote sensing a valuable tool to map the landscape and conduct large-scale spatial analysis to assess human impact over ecosystems and environmental evolution.(2)

Some studies have been conducted to map natural and artificial areas around the city of Bengaluru. Vanjare *et al.*(3) used images from the National Aeronautics and Space Administration (NASA) satellites Landsat 1 to 7 to evaluate urban growth in Bengaluru with a four-class land cover map (Built-up, Vegetation, Water, Barren land). Alex *et al.*(4) also made a four-class land cover map (Water, Barren land, Built-up, and Vegetation) from Landsat 7 images and GHS datasets (for Global Human Settlement, which are maps of built-up presence expressed as probabilities, made available using Sentinel Hub services) with a 30 m resolution in Bengaluru to assess green areas and built-up proportions over time. Ramachandra *et al.* used Landsat images to predict urbanization over time with IDRIS software and previous years' land covers as an input.(5) Similarly, Dehingia *et al.* produced a seven-class land cover classification map with Landsat, Indian Remote Sensing (IRS), and Sentinel-2A images to measure urbanization and predict it for 2030 and 2050.(6) In June 2022, Google released a paper on the "Dynamic World Model", a LULC deep learning model made from Sentinel-2 imagery aiming to make near real-time LULC parallel to Sentinel-2 acquisitions to cover various user needs. This model is a nine-class classification: Water, Trees, Grass, Crops, Shrub and Scrub, Flooded Vegetation, Built-Up, Bare Ground, and Snow & Ice.(7)

The authors of Refs. 2–6 mainly focused on urbanization assessment and temporal evolution. Moreover, the outputs of the presented pipelines do not match the targeted classes because they distinguish too few classes or provide too large resolutions (Landsat imagery provides a 30 m resolution at maximum). Indeed, we aim at 11 classes with a resolution of at least 10 m. Also, training data and LULC models for the previous papers are usually not easily downloadable. The "Dynamic World Model" distinguishes many different classes but only collects Sentinel-2 images and is unusable during the long monsoon period in India when cloud coverage is high.

Our work presents a novel pipeline for mapping multiclass land cover around Bengaluru using multitemporal and multispectral EO satellite images from multiple publicly available datasets. Using powerful free tools such as Google Earth Engine, Orfeo Toolbox, and machine learning algorithms, we ran all steps of an Object-based Image Analysis (OBIA) to produce an 11-class land cover map of more than 365,000 ha around Bengaluru. Using data from different sensors made it possible to overcome the limits raised by cloud coverage during the monsoon period. This method can be easily reproduced at a different time with cloud tools. The research objectives are as follows.

1. Produce an easily downloadable and up-to-date land cover map of the Bengaluru Agglomeration that covers all ground surface areas regardless of the cloud coverage at the time of data extraction.
2. Develop an easy to reproduce method to map the land cover around a fast-changing urban landscape city using open-source tools and open data.

2. Study Area and Materials

2.1 Study area

Bengaluru, the capital of the South Indian State of Karnataka, is located at 12.972N;77.595E and covers an area of 365,879 ha (Fig. 1). It comprises the urban center (Bengaluru urban) and the rural surrounding area (Bengaluru rural) of the city. In 2017, Ma *et al.* found that most supervised object-based land-cover image classification studies, areas of less than 300 ha (95.6% of the papers they reviewed) were investigated, which is a thousand times smaller than our working area.[8]

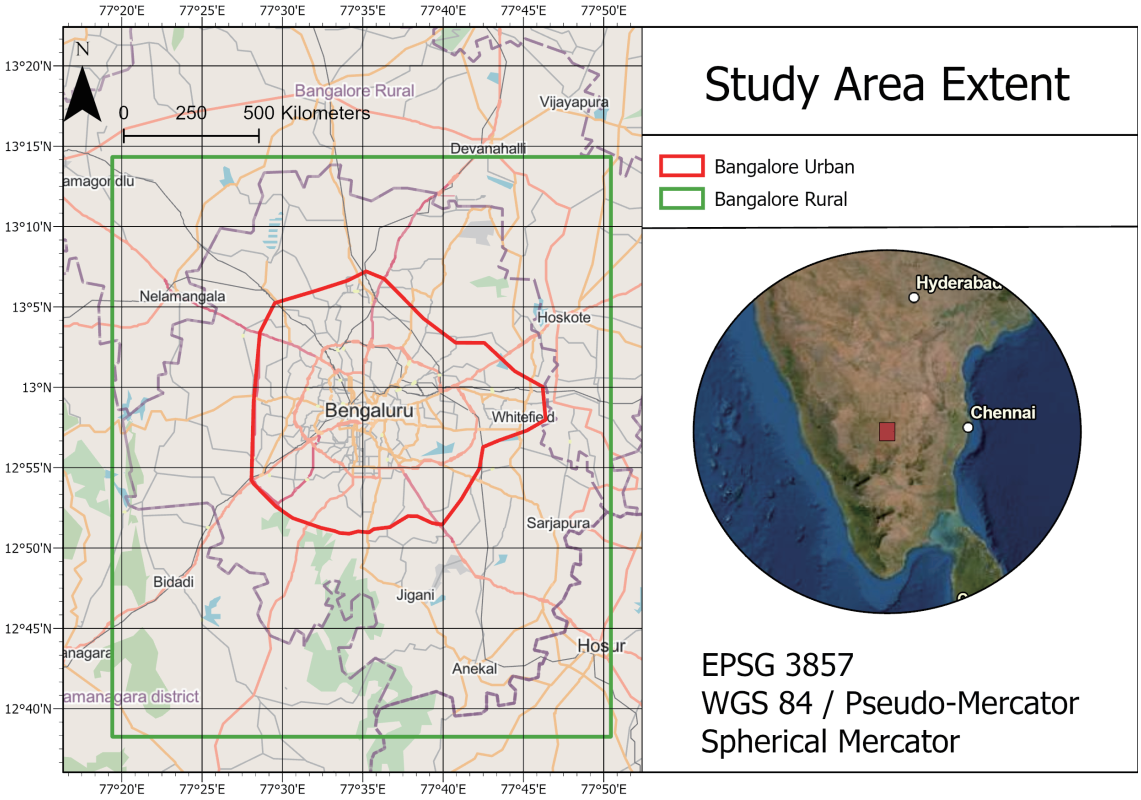


Fig. 1. (Color online) Map of Bengaluru study area and its location in the Karnataka region in India.

2.2 Materials

2.2.1 Data source

Here, the objective function F is computed for different values of the two parameters: spatial radius (*Hs*) and range radius (*Hr*), which are inputs. The two hypotheses that should be checked regarding the chosen data resolution are as follows.

(1) Objects of interest on the surface are larger than a pixel size.

(2) One pixel is composed of a single type of LULC.

The data used for this study are chosen to fit these hypotheses. They are from open access resources and are mainly from satellite imagery captured by remote sensing techniques. We use the European Sentinel-2 and Sentinel-1 datasets, the Indian Satellite CartoSat Digital Elevation Model (DEM) dataset, and data from the collaborative cartographic project OpenStreetMap. Sentinel-2 images are Level-2A with atmospherically corrected surface reflectance and extracted using the Google Earth Engine cloud platform. Nearly all bands are extracted from the Sentinel-2 dataset (from two satellites, 2A and 2B): Blue (2A:496.6 nm, 2B:492.1 nm), Green (2A:560 nm, 2B:559 nm), Red (2A:664.5 nm, 2B:665 nm), Red Edge 1 (2A:703.9 nm, 2B:703.8 nm), Red Edge 2 (2A:740.2 nm, 2B:739.1 nm), Red Edge 3 (2A:782.5 nm, 2B:779.7 nm), Red Edge 4 (2A:864.8 nm, 2B:864 nm), NIR (Near Infrared, 2A:835.1 nm, 2B:833 nm), SWIR1 (Shortwave Infrared 1, 2A:1613.7 nm, 2B:1610.4 nm), and SWIR2 (Shortwave Infrared 2, 2A:2202.4 nm, 2B:2185.7 nm). Extracted bands are rescaled to a 10 m resolution if necessary (The S2 bands used have an initial resolution of 10 or 20 m). Sentinel-2 images are composites extracted after applying cloud masks.

Sentinel-1 images are captured from a dual-polarization C-band synthetic aperture radar (SAR) instrument. Contrary to optic satellites, radar satellites have an active instrument that emits a wave toward its target and receives a new wave reflected by it. Commonly, radar waves can be transmitted and received in two polarizations: horizontal linear (H) and vertical linear (V). Since the scatter can change the polarization of the scattered wave compared with the original emitted wave, radar antennas are often made to emit and receive multiple polarizations. The instrument mode chosen is Interferometric Wide Swath (IW), which is available in single polarization (HH or VV) and dual polarization (HH + HV or VV + VH). Only VH and VV polarizations are kept since they are the only images of available areas all over Bengaluru under Sentinel-1 coverage. The resolution of Sentinel-1 images is 10 m. Only the “descending” orbit mode is available during our period of interest. The extraction is made using the Google Earth Engine cloud platform, and the preprocessing implements all steps from the Sentinel-1 Toolbox.(9)

The DEM from CartoSat was produced in May 2015 and extracted from the Bhuvan platform. OpenStreetMap data of roads are extracted for March 2022 with the QGIS plugin OSM. Sentinel-2 images, Google Earth WMS services, Google Street View, and local knowledge are used to create the ground truth for the training data.

2.2.2 Study area specificities

An important criterion for the choice of the extraction date is the climate of the Karnataka region. Bengaluru City experiences a tropical climate, which is classified as Aw Köppen's climate. This type of climate is characterized by a monthly temperature above 18 °C (22.9 °C in Bengaluru) and a distinct wintry dry season with less than 60 mm of rainfall per month (less than 50 mm for the Bengaluru dry season, which is from January to May).(10) In contrast, a monsoon period occurs from May to October with more than 125 mm of monthly rainfall. Naturally, the sky over Bengaluru is quite cloudy during this period, rendering the extraction of Sentinel-2 images difficult and justifying the use of available Sentinel-1 data regardless of the cloud cover.

Furthermore, extracting data to make an up-to-date LULC map justifies the choice of recent images. That is why Sentinel-2 images are extracted from January 2022 to May 2022 and Sentinel-1 images from January 2021 to June 2022. Sentinel-1 images were extracted during 2021 to obtain crop cycle information. We chose not to use Sentinel-2 data in 2021 because there were almost no available images during this period due to the large number of monsoons in 2021.

3. Methods

LULC classification techniques are powerful and versatile tools to analyze, assess, map, and monitor changes in the landscape structure. Technological advances in the acquisition of information from Earth's surface, algorithms to process them, and computational power have led to a quick augmentation of papers on the subject. Apart from deep learning approaches, there are two main approaches for LULC classification tasks: pixel-based classification and object-based classification. Pixel-based classification techniques classify each pixel in terms of its spectral value for different raster bands. Pixels from the same class are assumed to share similar spectral information. OBIA uses objects that are clusters of multiple homogenous pixels in terms of spectral information. It is a way of avoiding the "salt and pepper" effect of the pixel-based method caused by outlier pixels. It also provides the possibility to use textural information. Some groups also use a "subpixel-based approach," which considers clusters of pixels with a size between a pixel and an object.(2) The method used here is OBIA since targeted objects are multiple pixels wide, and it is considered a more robust method for potential outliers. The different steps of the method are Segmentation, Feature Extraction, and Classification. The pipeline is shown in Fig. 2.

3.1 Targeted classes

The targeted classes in this study are as follows.

- Built-up: Urbanized areas in the city center, villages in the urban part, or other artificial infrastructures (such as airports)
- Buildings: High structures with multiple apartments

- Croplands: Agricultural areas covered by nonperennial cultivations (divided into planted and nonplanted croplands between January and May 2022)
- Forest: Wild natural areas covered with trees
- Garden/Park: Enclosed land, partly wooded, designed for walking
- Plantation: Agricultural areas covered by perennial cultivations
- Shrublands: Wild areas that are mainly covered with shrubs, bushes, and sub-shrubs
- Roads
- Waterbody: Lakes, ponds, and rivers
- Drought wastelands: Drought and sterile areas

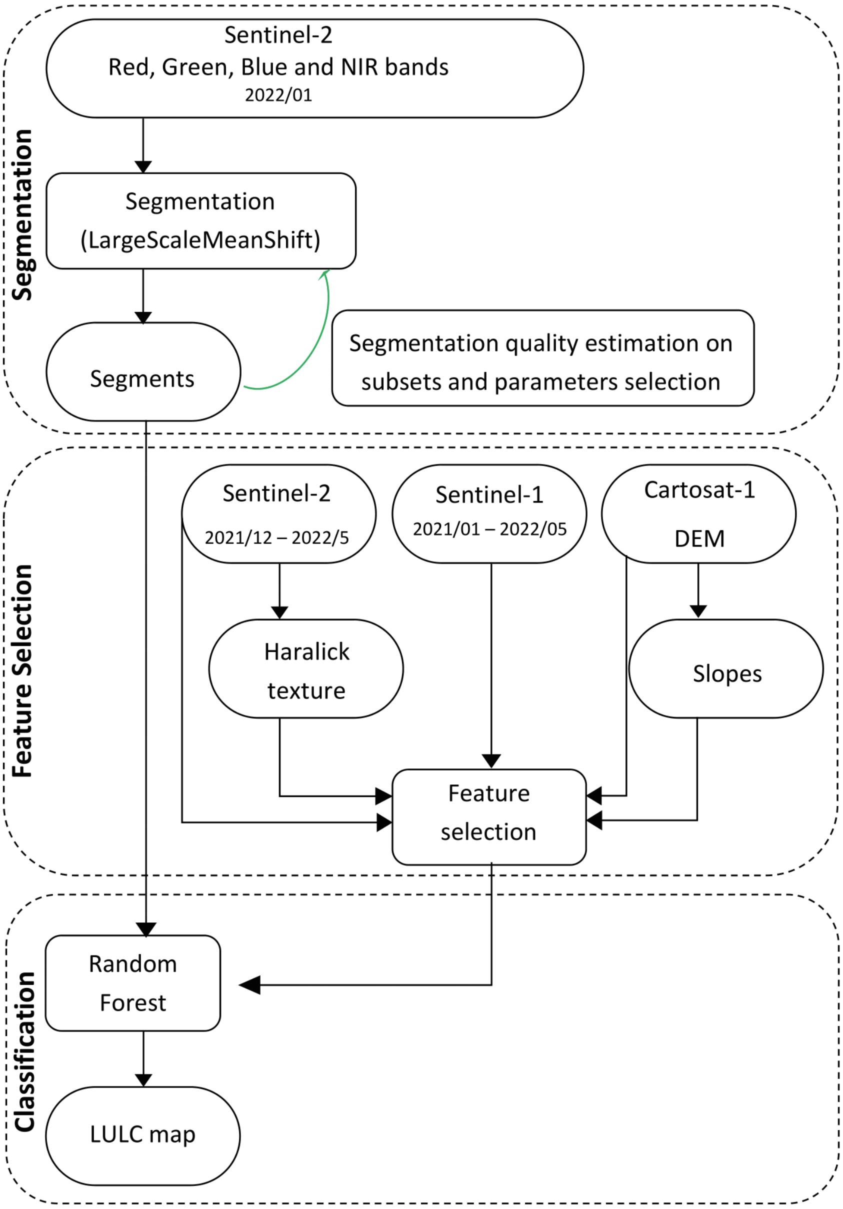


Fig. 2. (Color online) Full pipeline of the classification process.

3.2 Segmentation

OBIA classifies objects that are composed of multiple pixels. The first step of the method is called segmentation and consists in splitting the area into multiple objects, which are groups of pixels. These objects can also be called "segments". In our work, the segmentation is accomplished using a single-date raster with multiple bands. The following two constraints should be considered.

- Objects should contain pixels that are homogenous in terms of spectral information.
- Two contiguous objects should contain pixels that are heterogeneous in terms of spectral information.

Segmentation is a crucial step since the quality of feature extraction and classification highly depends on the precision of the segmentation.(11) If an object from segmentation is too large and thus made of two classes, it will confuse the classification algorithm. With very small objects, misclassifications due to outlier values may occur since one pixel's value has more importance in the classification. Ideally, segmentation objects should be the same size as the objects we want to detect. Segmentation methods can be categorized into spectrally based methods, spatially based methods, hybrid methods, and semantic methods.(12) Here, we use the Region Growing method MeanShift whose algorithm LSMS is made available by the open-source software Orfeo Toolbox (OTB). It chains the four steps of the MeanShift framework: MeanShiftSmoothing, LSMSSegmentation, LSMSSmallRegionsMerging, and LSMSVectorization.

Most OBIA studies do not use a segmentation accuracy assessment method to calibrate the segmentation parameters and only perform qualitative visual assessments.(13) However, we consider that this step has a crucial effect on the final classification quality. With a poorly performing segmentation, two objects from different classes can be segmented as one single object, which will confuse the classification algorithm. Currently, there are two types of method to assess the segmentation quality: supervised and unsupervised approaches.(12) Here, we use the unsupervised segmentation assessment method from *Parameter selection for region-growing image segmentation algorithms using spatial autocorrelation*.(14) We compute two indexes, the intersegment variance α (measuring the homogeneity of the intersegment pixel values) and Moran's I autocorrelation index (measuring the heterogeneity of the intersegment pixel values):

$$v = \frac{\sum_{i=1}^{n} a_i v_i}{\sum_{i=1}^{n} a_i}, \tag{1}$$

where a_i and v_i are the area and intra-object variance of segment i, respectively.

$$I = \frac{n}{S_0} * \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{i,j} z_i z_j}{\sum_{i=1}^{n} z_i^2}, \tag{2}$$

where z_i is the deviation of the brightness value of object i from the mean of all objects $(x_i\text{-}\bar{x})$, $w_{i,j}$ is the spatial weight between objects i and j, which is one for adjacent regions or zero otherwise, n is the total number of objects, and S_0 is the aggregate of all spatial weights.

$$S_0 = \sum\nolimits_{i=1}^{n} \sum\nolimits_{j=1}^{n} w_{i,j} \tag{3}$$

The goal is to minimize α and β. Then, the objective function is given by $F(v, I) = F(v) + F(I)$, where $F(v)$ and $F(I)$ are normalization functions.

$$F(x) = \frac{x_{max} - x}{x_{max} - x_{min}} \tag{4}$$

Here, the objective function F is computed for different values of the two parameters, spatial radius (*Hs*) and range radius (*Hr*), which are inputs of the LSMS algorithm. *Hs* is the spatial limit above which two pixels are not merged. *Hr* is the maximum distance between two pixels in terms of band values above which two pixels are not merged. The segmentations are made with red, green, blue, and infrared bands of a Sentinel-2 composite image taken in January 2021. Random square patches of the image are selected for segmentation assessment. The proportion of the area used in segmentation assessment is the same as in the original paper. This date is chosen because it is the only date when we were able to extract a raster without any cloud coverage. δ is calculated for each segment from the output of LSMS for the January 2021 grayscale raster. In particular, α is computed using the Python packages *geopandas* and rasterstats and β with the Python package *PySAL*. The minimum size parameter of LSMS is set to 10000 m^2 (which corresponds to the area of an object made up of 100 Sentinel-2 pixels) since it is the size of the most miniature targeted objects, namely, small lakes of 10000 m^2. *Hs* is tested in an interval of 2 to 40 with a step of 2 and *Hr* in an interval of 2 to 30 with a step of 2. Finally, after segmentation, roads are clipped from the segmentation output. Roads are originally line vectors extracted from the OpenStreetMap open-source API. Five types of roads are then converted to polygons using the QGIS built-in Buffer function with a buffer distance depending on the type of road. The chosen buffer diameters are as follows: 40 m for Highway motorway, 30 m for Highway trunk, 20 m for Highway primary, 20 m for Highway secondary, 10 m for Highway tertiary, 10 m for Highway residential, and 10 m for Highway unclassified. The presented road nomenclature is taken from OpenStreetMap.

3.3 Feature extraction

Feature extraction is the process of collecting traits that are unique to each object and selecting the most meaningful of them to discriminate and classify them into the targeted classes. A feature is a numeric representation of an aspect of the raw data. This classification step aims to extract features from raw data and make new combinations that are meaningful for the classification machine learning model (this step is called feature engineering). Then feature selection aims at filtering the most meaningful features: too many features compared with the number of training samples leads to difficult learning and sometimes impossible model convergence to a proper classification scheme. All the features' values extracted in this part are extracted at the pixel scale; then, summary statistics are calculated at the object scale. The latter are computed with the OTB ZonalStatistics algorithm. This algorithm takes a raster and a polygon shapefile to extract the maximum, mean, standard deviation, and minimum of each band of the raster for all segments.

3.3.1 Feature engineering

All bands from Sentinel-2 images from the most recent available images are extracted: Red, Green, Blue, NIR, RE1, RE2, RE2, RE4, SWIR1, and SWIR2 for monthly composites from December 2021 to May 2022. Then neochannels (which are new bands whose formula is made up of multiple bands) are computed from raw Sentinel-2 bands for each date:(15–17)

$$NDVI = \frac{NIR - Red}{NIR + Red} \tag{5}$$

$$NDWI = \frac{SWIR - NIR}{SWIR + NIR} \tag{6}$$

$$BSI = \frac{(SWIR2 + Red) - (NIR + Blue)}{(SWIR2 + Red) + (NIR + Blue)} \tag{7}$$

$$MBI = \frac{SWIR1 - SWIR2 - NIR}{SWIR1 + SWIR2 + NIR} + f \tag{8}$$

Neochannels are computed using the Google Engine cloud platform. The neochannels are chosen for their ability to distinguish classes that seemed to have similar spectral signatures in preliminary tests. Values of the Cartosat-1 DEM are also extracted. From DEM, slopes are computed with the QGIS built-in slope tool. Moreover, some of the Haralick textures are extracted from the grayscale raster of January 2022.(18) Texture can be defined as a spatial organization of physical elements showing a repeated pattern robust to translation, although there is considerable spectral variability. The Gray Level Co-occurrence Matrix (GLCM) is computed with a 3 × 3 size sliding kernel. Among all available texture features, eight are computed using the *HaralickTextureExtraction* function from OTB in the QGIS environment: Energy, Entropy, Correlation, Inverse Difference Moment, Inertia, Cluster Shade, Cluster Prominence, and Haralick Correlation. Finally, VH and VV polarizations are extracted from Sentinel-1 datasets between January 2021 and June 2022 in the form of monthly composites. Since Sentinel-1 data are available every month without the constraint of monsoons, we extracted information within a wider period than data from Sentinel-2 to capture crop dynamics. A neochannel is also computed:(19)

$$RVI = \frac{4\delta_{VH}}{\delta_{VV} + \delta_{VH}}. \tag{9}$$

All these indexes were proven to be able to distinguish between classes with close spectral signatures. For example, NDVI can help in discriminating classes of different vegetation types.

3.3.2 Feature selection

Finally, 404 features are extracted from raw rasters and ε to θ. Fitting a machine learning model with too many features might cause the problem of the curse of dimensionality: having too many features compared with the training sample size (Fig. 3) implies a risk of overfitting for the model, which will incur difficulties in understanding borders between classes. Regrouping samples into classes needs a lot of training data to clearly define the borders. Otherwise, we will have a poor accuracy with real data. Working with high-dimensional data may make the search of patterns more difficult. A less complex model is also easier to interpret and quicker to train. To solve this problem, feature selection methods are used. The first method to remove some useless features is to use correlations between features. Correlation is a statistical technique measuring how one variable relates to another. It is a bivariate analysis that shows how the change of one variable is linked to the variations of the other. The correlation coefficient computed here is the Pearson correlation, which is defined as

$$r = \frac{Cov(X,Y)}{\delta_X \delta_Y}. \tag{10}$$

On the basis of pairwise correlations with a threshold of 0.90 (positive or negative), we can delete one of the two correlated features in each pair. Then, the wrapper selection method, named forward selection, from the Python package Scikit-learn is applied.(20) It is an iterative and greedy method that evaluates model performance when adding each feature one at a time.

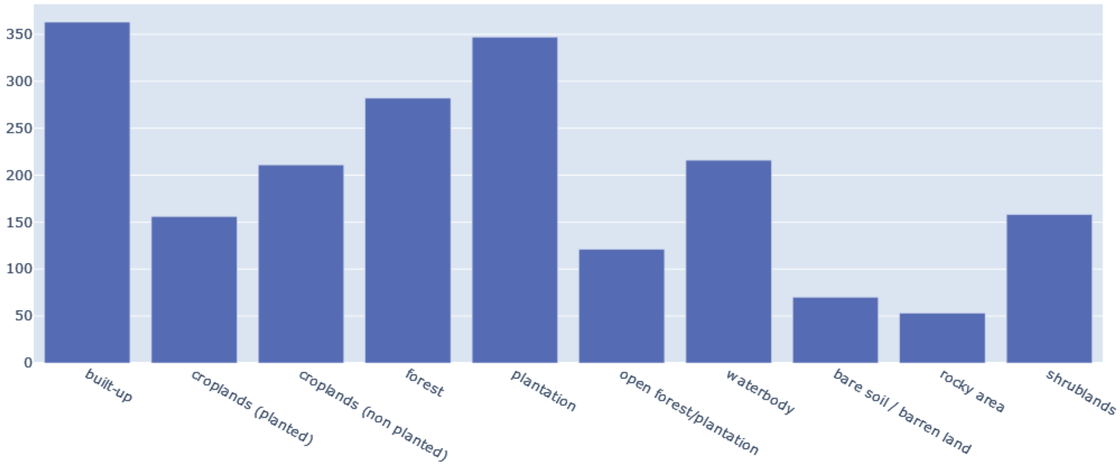


Fig. 3. (Color online) Sizes of classes in training sets.

Each addition is made considering the lowest p-value among features. Finally, PCA could be used to reduce the feature space dimension but does not prove effective in our case. We trained the model for different sizes of the feature set: 10 to 50 with a step of 10. $F1$ is the metric used to evaluate the performance for each number of features. $F1$ for one class is defined as follows.

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$
$$\text{with } precision = \frac{tp}{tp + fp} \tag{11}$$
$$\text{and } recall = \frac{tp}{tp + fn}$$

tp is the number of objects classified to the targeted class that they really belong to, fp is the number of objects classified to the targeted class that they do not belong to, and fn is the number of objects that are classified in a class other than the one targeted, but which, in fact, belong to the targeted class.

3.4 Classification

We train three easy-to-use models widely employed for classification: Support Vector Machine (SVM),[20] Random Forest,[21] and the Gradient Boosted classification algorithm LightGBM.[21–23] Hyperparameters were set to default values, and the global $F1$ score was used as the classification accuracy metric. Classification algorithms compared in this work have been widely used in remote sensing papers. Classification pipelines are implemented using the Python packages Scikit-Learn and Yellowbrick. Training is done using the wrapper method, first, to define the best features for each model and for different feature set sizes, then run on the best subset of features. One important criterion for training is the number of training samples and the number of samples per class. Unbalanced classes should be avoided since large classes would affect the model considerably. Moreover, small classes with poor accuracy will not greatly influence global accuracy and go unnoticed. Models will have difficulties generalizing on small classes. Thus, following two constraints must be considered.

- Having enough training samples
- Having balanced classes

To create the training samples, the grid produced for segmentation is used. One hundred sixty-two cells are selected, and the objects from segmentation in them are annotated. Two thousand one hundred and eighty samples have been annotated (2% of the total number of objects after segmentation). Proportions of each class are presented in Fig. 3. The respective proportions are representative of real surface frequency. Some classes are smaller owing to the lack of real surfaces in the chosen annotated areas.

The best subsets of features are trained through fivefold cross-validation. It consists of training the model multiple times on different data splits. Each split is composed of a training and a testing set. For each split, the model is trained on the training set and evaluated on the testing set. This technique avoids splitting of the data that would not be representative of the global data. Accuracies reported in this paper are means over the whole fivefold cross-validation process. In addition to that, stratified sampling, which is a sampling method applied to data before allocating them to the folds, is applied to the splitting. Its role is to produce testing and training samples that have the same proportions of each class as in the original data. By applying stratified sampling and *k*-fold cross-validation, we ensure that each training–testing sampling pair is representative of the original set of training data and that training accuracy is not affected by outlier training sets.

3.5 Accuracy assessment

The model's accuracy is assessed at the object scale, which means that the accuracy is computed depending on the class allocated to the objects. The main accuracy assessment metric used in this work is the $F1$ score. Global $F1$ scores are used for features and model selections, and the per-class $F1$ score is used to obtain detailed results for each class using the best model. *Precision* and *recall* metrics can also provide information about the classification. The use of the confusion matrix also gives details about which classes are difficult to distinguish.

4. Results

4.1 Segmentation

The results of the segmentation are presented in Fig. 4. The maximum of the objective function F, which means low intra-object heterogeneity and high inter-object heterogeneity is reached for a value of 26 for Hs and 10 for Hr. Values of F can vary considerably between two close points, which might show that either the method is not robust and causes randomness since

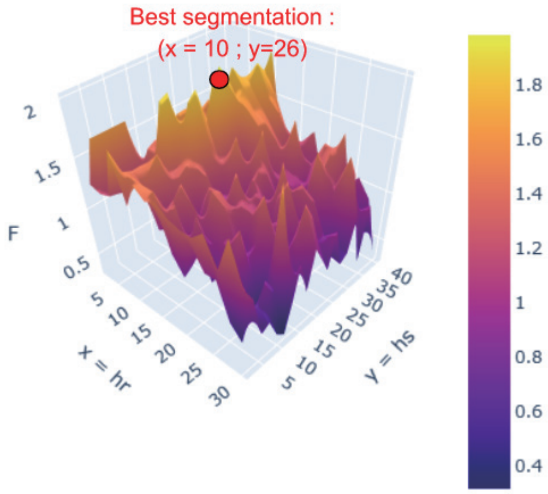


Fig. 4. (Color online) Values of the Segmentation Objective Function for all parameter pairs.

two close segmentations can be very different, or the differences between two segmentations are small in absolute values. However, there is a real tendency of a global increase in F along the Hr axis. We can find out visually that the Hr parameter affects the value of the objective function much more than Hs. Indeed, the Pearson correlation coefficient between Hr and F is −0.64 with a p-value of 2.41^{-36}, whereas it is 0.12 with a p-value of 0.03 between Hs and F. This indicates that it will not be necessary to compute all (Hs, Hr) pairs if the method is reproduced, which will save much computational time. On the basis of the results of the segmentation, the segmentation with the parameters $Hr = 10$ and $Hs = 26$ is chosen. An example of the segmentation outputs is shown in Fig. 5.

4.2 Feature selection

Results for each model are presented in Figs. 6 and 7. $F1$ scores output from the k-fold cross-validation for each combination are close. Output scores are often over 0.80, which means that the feature selection method successfully filtered hundreds of features to keep a few dozen that are meaningful and sufficient to distinguish all classes. Further tests with more features did not show a significant improvement in accuracy. Adding even more features again will ultimately decrease accuracy, since, when we trained the model with all features, we obtained a very low accuracy (global $F1$ score < 0.50).

The best combination selected by the method is Random Forest using 40 features. Scores are very close, and we chose to select the best global $F1$ score.

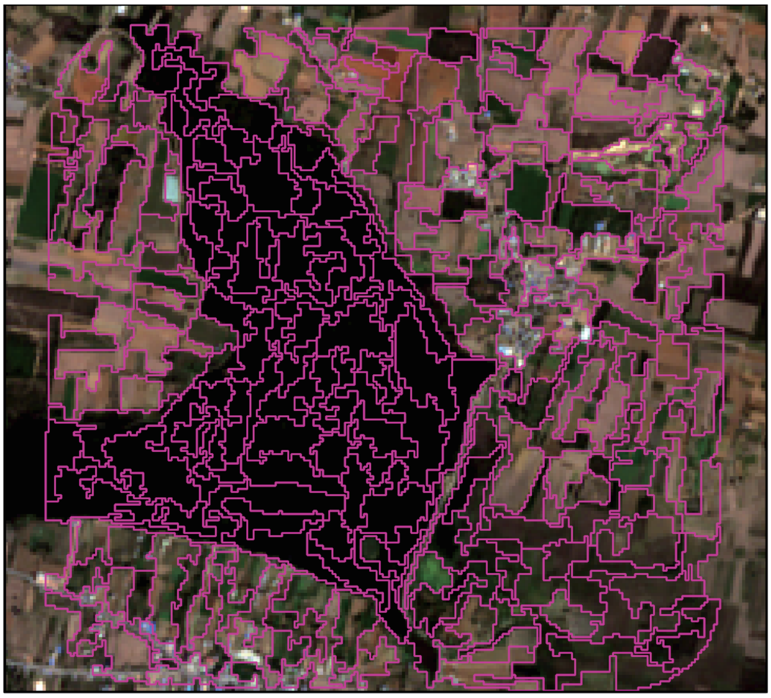


Fig. 5. (Color online) An example of segmentation outputs.

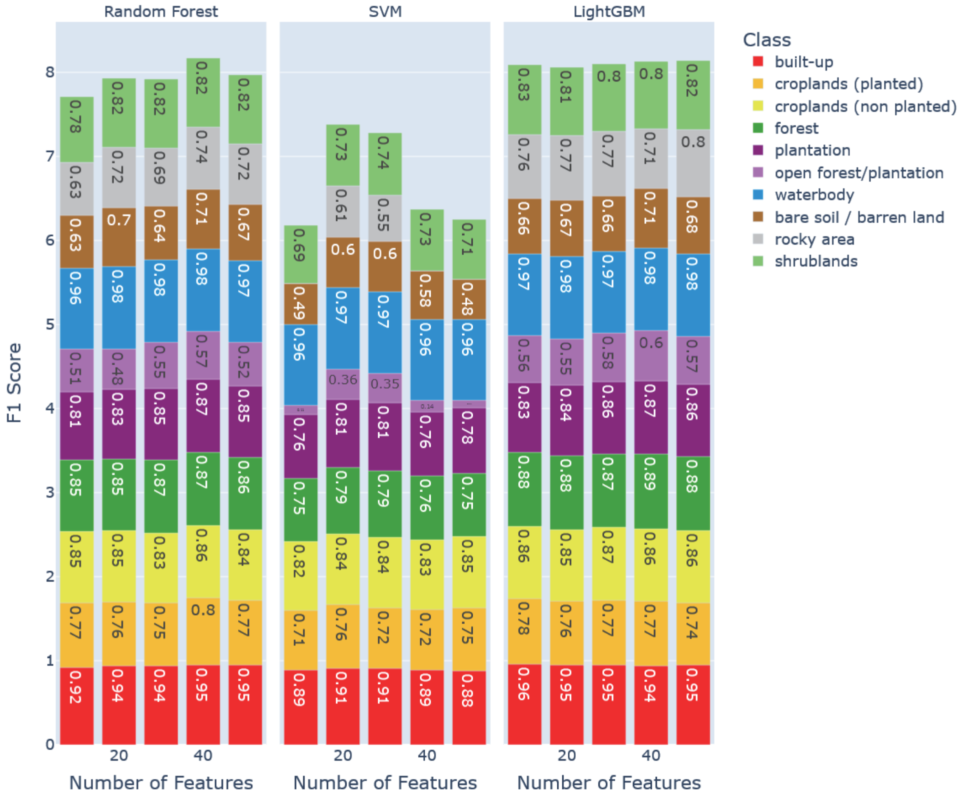


Fig. 6. (Color online) *F*1 score for all classes with each combination of model and number of features.

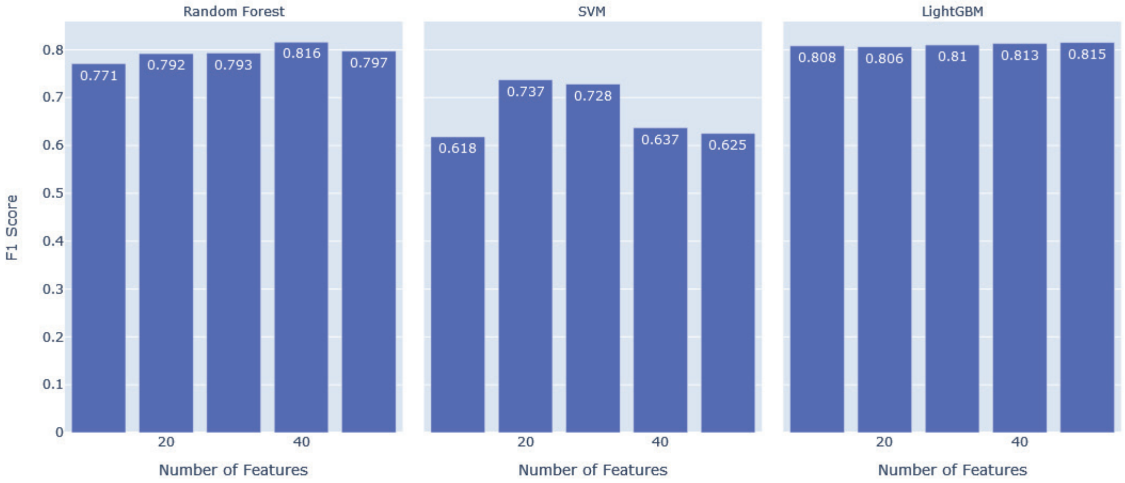
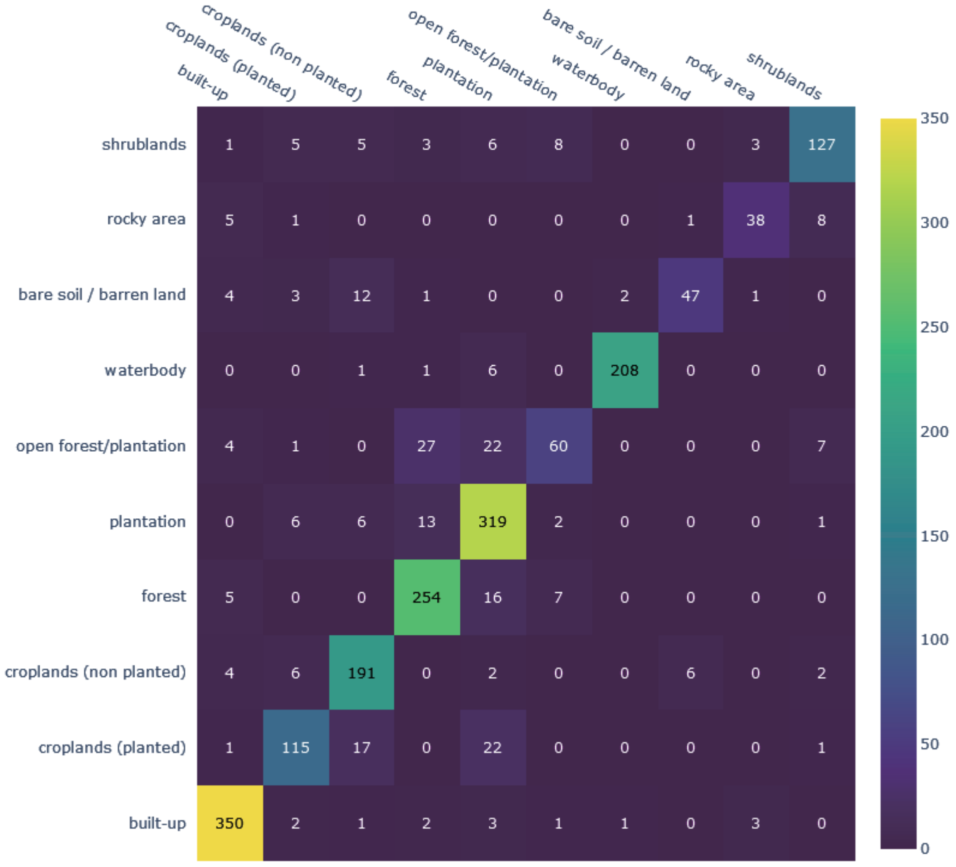


Fig. 7. (Color online) Global *F*1 score with each combination of model and number of features.

Among the best selected features, EVI and BSI are considered interesting in many months (January, March, April, and May). Since these indexes are designed to monitor vegetation density, we can assume that they are selected to quantify vegetation differences between the beginning and the end of the summer period to detect crops that had been planted in contrast to croplands with no crops cultivated during summer and the ground without any plants. NDVI is also selected for May, probably partly for the same goal. Sentinel-1 VH, VV (copolarized and cross-polarized backscattered data collected from Sentinel-1 data), and the Radar Vegetation Index (RVI) are also selected for multiple months and confirm this analysis. EVI, NDVI, and Sentinel-1 data must also help to distinguish plantations and forests. The SWIR2 (shortwave infrared 2) band from Sentinel-2 is selected in April 2022. This could help to distinguish rocky areas.(24) Slope and DEM features are selected, which suggest that some types of surface are more represented in certain heights, altitudes, and steepness. All selected indexes are listed in Appendix 1.

4.3 Classification

The best algorithm with the best subset of features are kept for final classification. The best combination is the Random Forest algorithm with 40 features from Sentinel-2 images, Haralick features, DEM, slopes, and Sentinel-1 images. The Random Forest is trained on all training data using a stratified fivefold cross-validation method. The confusion matrix of the training step is presented in Fig. 8. It is the sum of all confusion matrices from the *k*-fold training. Most of the training samples are in the diagonal, meaning they are well classified in the process. The



	built-up	croplands (planted)	croplands (non planted)	forest	plantation	open forest/plantation	waterbody	bare soil / barren land	rocky area	shrublands
shrublands	1	5	5	3	6	8	0	0	3	127
rocky area	5	1	0	0	0	0	0	1	38	8
bare soil / barren land	4	3	12	1	0	0	2	47	1	0
waterbody	0	0	1	1	6	0	208	0	0	0
open forest/plantation	4	1	0	27	22	60	0	0	0	7
plantation	0	6	6	13	319	2	0	0	0	1
forest	5	0	0	254	16	7	0	0	0	0
croplands (non planted)	4	6	191	0	2	0	0	6	0	2
croplands (planted)	1	115	17	0	22	0	0	0	0	1
built-up	350	2	1	2	3	1	1	0	3	0

Fig. 8. (Color online) Confusion matrix from classification process.

training results in an overall accuracy of 0.846. As expected in the explanatory study, most of the confusion arises between vegetation classes. Twenty croplands that had been planted in 2022 are misclassified as nonplanted ones and 17 of them as plantations. Concerning the former, it could be due to training samples in the early stage of the plantations that have a spectral signature between planted and nonplanted crops. The latter could be the result of the confusion between croplands and sparse plantations. Misclassifications are also made for drought wastelands that are classified as nonplanted croplands. This must be because these two types of soil are physically similar. Moreover, 15 plantations are classified as forests and 16 forests as plantations. The amount of misclassifications between forests and plantations is small, considering the difficulty in distinguishing these two classes estimated during the preliminary analysis. Finally, 47% of the open forest/plantation are misclassified into Forest or Plantation. This is probably due to the similarity of spectral signatures between these classes. More importantly, the resolution of Sentinel-2 images makes it difficult to estimate vegetation density compared with more precise data. Also, the class open forests/plantations were qualitatively defined and may lack a quantitative definition.

Recall, *Precision*, and *F*1 score metrics for all classes are presented in Fig. 9. *Precision* is, for one class, the number of correctly classified objects among all objects that are classified in it. *Recall* is, for one class, the number of correctly classified objects among all objects that should have been classified in it. When *precision* is high, it means that most objects in a class belong to it. When *recall* is high, it means that most objects that really belong to a class are really classified in it. *Precision* is greater than *recall* for planted croplands, open forests/plantations, water bodies, drought wastelands, rocky areas, and shrublands, which means that these classes capture fewer objects than the others, but when they are classified, they have a good chance of being well classified. However, some objects that should be classified in it are not. For the other classes, most of the objects that really belong in them are well classified, but they may have many false positive classifications. Open forests/plantations have a very low recall, meaning that many areas considered as belonging to them are classified in other classes. To a slightly lesser extent, planted croplands, drought wastelands, and rocky areas also have low recall. These results have a negative impact on the *F*1 score, which is very low for open forest/plantations. This class's

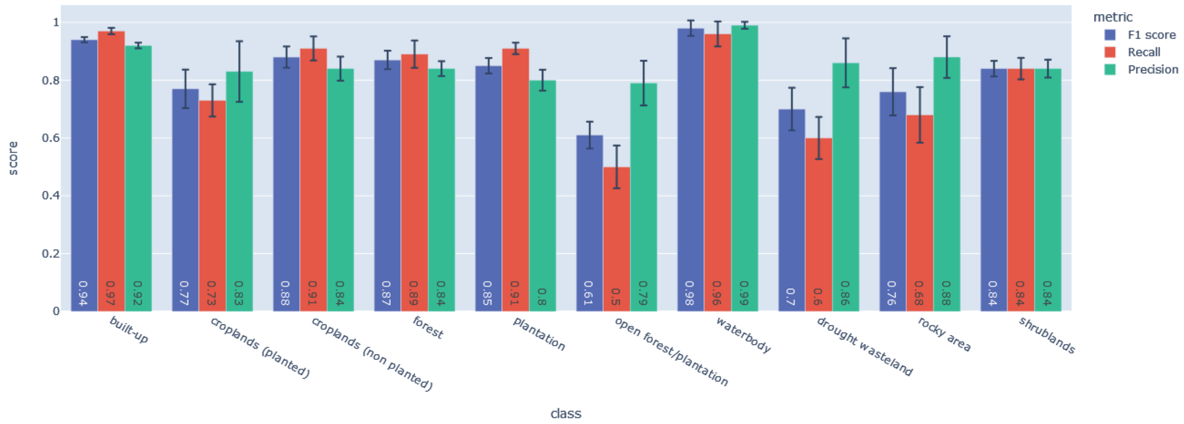


Fig. 9. (Color online) *Recall*, *Precision*, and *F*1 score for each class from classification process.

recall barely reaches 40%, meaning that half of the actual open forest/plantations are misclassified. The final classification run over the whole area with the model trained on all examples is presented in Fig. 10.

5. Discussion

The study area is unusually larger than those used in standard remote sensing papers.[8] This constraint requires many training samples to be overcome. Despite the limited size of the training set, the model achieves good accuracy. In particular, the crops, plantations, and forest classes are well distinguished. However, the classification accuracies of rocky areas and open forests/plantations are low owing to their being confused with forests or shrublands and built-up or shrublands, respectively. The accuracy of open forests/plantations probably suffers from the low resolution of the data that does not correctly reflect the vegetation density. High-resolution data paired with high-resolution textures could solve this problem. Moreover, whereas open forests and plantations were only defined as tree areas with openings, a quantitative definition with a percentage of tree coverage should be explored in the future. For now, we could also merge "forests" with "open forests" and "rocky areas" with "shrublands" to achieve higher accuracy. More training samples for these classes that are not the most numerous could also help to tackle this problem.

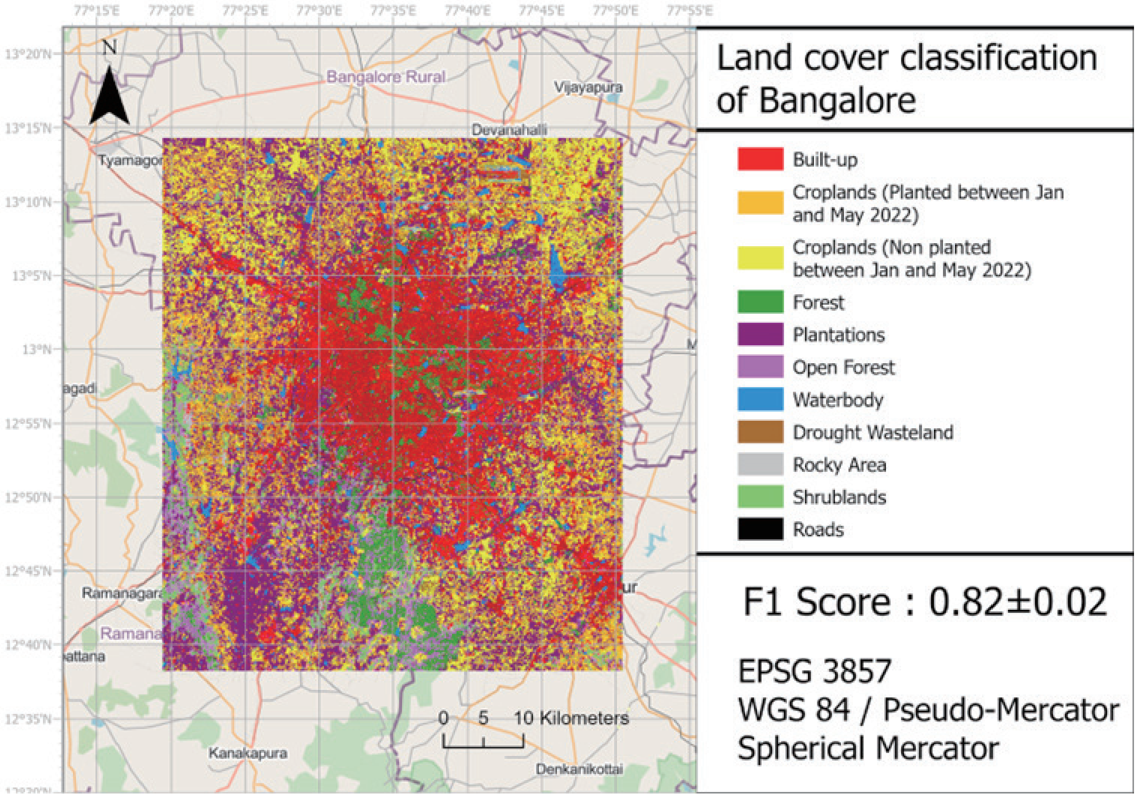


Fig. 10. (Color online) Land cover classification of Bengaluru.

Furthermore, since classified objects are outputs of segmentation, classification accuracy highly depends on it. Parameter selection methods are scarce for segmentation, and the one used in this study only allows the mobilization of four Sentinel-2 bands: Red, Green, Blue, and NIR for January 2022. Results of the segmentation part, and, by extension, results of the classification, could be enhanced with the use of more bands or indexes (such as the vegetation index to better cluster spatial vegetation patches, for example). Whereas different parameters with the same bands only correct some object breakdowns, the use of several more bands through PCA might increase the precision of the results for this step. Indeed, in the final segmentation, we can see some objects are wrongly bound and cause uncertainties during the annotation step and complex classification. This problem would not be solved using a different model or a model with better parameterization.

Concerning the annotation step, we assume that we can gain a few accuracy percentage points with more training samples. Another obstacle is the availability of good satellite images for the period of interest and the lack of terrain knowledge. Ground truth was not made by people on the ground in India. Images for annotation validation were made using Sentinel-2 images and Google Earth's high-resolution images. Since the latter were not the latest up-to-date images, we could not use them for crop states, for example. Moreover, images were insufficient from June because of the occurrence of monsoons. Using Google Street View, Sentinel-2 images, and local knowledge with Indian researchers, a satisfactory training set was made. However, a ground truth training set made by people in the field using high-resolution and up-to-date data could bring more assurance to its validity. It could also make it possible to distinguish crop and plantation species and determine a more precise land cover with more classes.

However, the training results and the model resolution should be sufficient to map nesting and foraging sites because the foraging location that is decoded from the study of the waggle dances of *Apis dorsata* is on the order of meters and beyond.(25) Thus, a precision higher than a meter would not make sense. Some studies have succeeded in showing the tendencies of foraging places with less classes and less precise land use land cover maps.(26) In urban and peri-urban spaces, honey bees nest near and forage in large green areas, small parks, and small green patches as well.(27) Our classification will surely be sufficient to detect most foraging areas that could help local groups to estimate places important to *Apis dorsata* colonies. Nevertheless, we could not clearly capture all the landscape details. After detecting the most visited places, it could be relevant to make another classification with better resolution.

6. Conclusion

This research was carried out to produce the most recent 11-class and 10-m precision land cover map of the complete metropolitan area of the tropical megacity Bengaluru. Land cover maps of such large areas have only rarely been produced so far. Moreover, the proposed pipeline uses only open data and cost-free alternatives. Researchers could reuse this pipeline to create up-to-date maps in the future. We incorporated three machine learning algorithms: Random Forest, SVM, and LightGBM. Notably, Random Forest emerged as the top performer, achieving an impressive global $F1$ score of 0.82. In our analysis, we utilized training data from multiple sensors, spanning January 2021 to June 2022, and a DEM from 2015. These data, both in their raw form and as engineered features (including EVI and RVI), were employed to train our classification model. Leveraging feature selection techniques, we identified the 40 most informative features for classification. Our exploration of data from multiple sensors proved pivotal in addressing a significant challenge—overcoming dense cloud cover associated with annual monsoons, which traditionally hindered optical image classification. We believe that the availability of large-scale, easily generated land cover maps will be instrumental in future endeavors to assess the impacts of escalating urbanization on conflicts of interest among humans and the associated threats to the environment and biodiversity.

Acknowledgments

This project was part of the initiative "Exploring and managing human–bee conflict in Asian cities using AI" supported by Google in the workshop "AI for social good.". The aim of this workshop was to bring a positive impact of AI to communities that had not benefited from it in the past.

References

1 W. Akram, A. Sajjad, M. Ali, A. Ahmad, I. Ali, B. Saddiq, M. Yasin, and M. A. Aqueel: J. Asia-Pacific Entomology **25** (2022) 101958. https://doi.org/10.1016/j.aspen.2022.101958

2 L. S Macarringue, É. L. Bolfe, P. R. M. Pereira: J. Geogr. Inf. Syst. **14** (2022) 1. https://doi.org/10.4236/jgis.2022.141001

3 A. Vanjare, S. N. Omkar, and J. Senthilnath: Int. J. Image, Graph. Signal Process **6** (2014) 18. https://doi.org/10.5815/ijigsp.2014.10.03

4 E. C. Alex, K. V. Ramesh, and H. Sridevi: Int. J. Civil Eng. Technol. **8** (2017) 597.

5 T. V. Ramachandra, H. Bharath Aithal, S. Vinay, N. V. Joshi, U. Kumar, and K. V. Rao: 30th Annual In-House Symp. Space Science and Technology, ISRO-IISc Space Technology Cell, Indian Institute of Science, Bangalore (2013) 7–8.

6 H. Dehingia, R. R. Das, S. A. Rahaman, P. Surendra, and A. D. Hanjagi: Int. Archives Photogramm., Remote Sens. Spatial Inf. Sci. **43** (2022) 589. https://doi.org/10.5194/isprs-archives-XLIII-B3-2022-589-2022

7 C. F. Brown, S. P. Brumby, B. Guzder-Williams, T. Birch, S. Brooks Hyde, J. Mazzariello, W. Czerwinski, V. J. Pasquarella, R. Haertel, S. Ilyushchenko, K. Schwehr, M. Weisse, F. Stolle, C. Hanson, O. Guinan, R. Moore, and A. M. Tait: Sci. Data **9** (2022) 251. https://doi.org/10.1038/s41597-022-01307-4

8 L. Ma, M. Li, X. Ma, L. Cheng, P. Du, and Y. Liu, : ISPRS J. Photogramm. Remote Sens. **130** (2017) 277. https://doi.org/10.1016/j.isprsjprs.2017.06.001

9 L. Veci, P. Prats-Iraola, R. Scheiber, F. Collard, N. Fomferra, and M. Engdahl: Proc. IEEE Int. Geoscience Remote Sensing Symposium (IGARSS, IEEE) (2014) 1–3.

10 M. Kottek, J. Grieser, C. Beck, B. Rudolf, and F. Rubel: World map of the Köppen-Geiger climate classification updated (2006). https://doi.org/10.1127/0941-2948/2006/0130

11 M. D. Hossain and D. Chen: ISPRS J. Photogramm. Remote Sens. **150** (2019) 115. https://doi.org/10.1016/j.isprsjprs.2019.02.009

12 I. Kotaridis and M. Lazaridou: ISPRS J. Photogramm. Remote Sens. **173** (2021) 309. https://doi.org/10.1016/j.isprsjprs.2021.01.020

13 S. Ye, R. G. Pontius Jr, and R. Rakshit: ISPRS J. Photogramm. Remote Sens. **141** (2018) 137. https://doi.org/10.1016/j.isprsjprs.2018.04.002

14 G. M. Espindola, G. Camara, I. A. Reis, L. S. Bins, and A. M. Monteiro: Int. J. Remote Sens. **27** (2006) 3035. https://doi.org/10.1080/01431160600617194

15 B. C. Gao: Remote Sens. Environ. **58** (1996) 257. https://doi.org/10.1016/S0034-4257(96)00067-3

16 C. T. Nguyen, A. Chidthaisong, P. K. Diem, and L. Z. Huo: Land **10** (2021) 231. https://doi.org/10.3390/land10030231

17 N. Pettorelli, J. O. Vik, A. Mysterud, J. M. Gaillard, C. J. Tucker, and N. C. Stenseth: Trends Ecol. & Evol. **20** (2005) 503.

18 R. M. Haralick, K. Shanmugam, and I. H. Dinstein: IEEE Trans. Syst., Man, Cybernetics **6** (1973) 610. https://doi.org/10.1109/TSMC.1973.4309314

19 H. Le Minh, T. V. Van, and T. T. Anh: Proc. 2019 40th Asian Conf. Remote Sensing (ACRS 2019, Daejeon, Korea) pp. 14–18.

20 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay: J. Mach. Learn. Res. **12** (2011) 2825.

21 W. S. Noble: Nature Biotechnol. **24** (2006) 1565. https://doi.org/10.1038/nbt1206-1565

22 M. Pal: Int. J. Remote Sens. **26** (2007) 217. https://doi.org/10.1080/01431160412331269698

23 G. Ke., Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Qi. Ye, and T. Liu: Adv. Neural Inf. Process. Syst. **30** (2017).

24 C. Laukamp, A. Rodger, M. LeGras, H. Lampinen, I. C. Lau, I, B. Pejcic, J. Stromberg, N. Francis, and E. Ramanaidou: Minerals **11** (2021) 347. https://doi.org/10.3390/min11040347

25 R. Schürch, K. Zwirner, B. J. Yambrick, T. Pirault, J. M. Wilson, and M. J. Couvillon: Animal Behav. **150** (2019) 139. https://doi.org/10.1016/j.anbehav.2019.01.016

26 I. Steffan-Dewenter, and A. Kuhn: Proc. Royal Soc. London. Series B: Biol. Sci. **270** (2003) 569. https://doi.org/10.1098/rspb.2002.2292

27 M. Garbuzov, R. Schürch, and F. LRatnieks: Urban Ecosyst. **18** (2015) 411. https://doi.org/10.1007/s11252-014-0403-y

Data Availability Statement

The land cover map resulting from this study is openly available from bengaluru_lulc_2022-dataset at https://github.com/UTokyo-FieldPhenomics-Lab.

Appendix 1

Selected features for the best classification:

'S2_2021-12_MBI_Stdv',
'S2_2021-12_NDVI_Mean',
'S2_2021-12_RE1_Mean',
'S2_2021-12_SWIR1_Mean',
'S2_2022-1_MBI_Mean',
'S2_2022-1_RE1_Mean',
'S2_2022-1_BSI_Stdv',
'S2_2022-1_NDWI_Stdv',
'S2_2022-1_blue_Stdv',
'S2_2022-2_NDWI_Mean',
'S2_2022-3_MBI_Mean',
'S2_2022-5_NDVI_Mean',
'S2_2022-5_RE4_Mean',
'S2_2022-5_BSI_Stdv',
'S2_2022-5_EVI_Stdv',
'S2_2022-5_NDWI_Stdv',
'S2_2022-5_RE4_Stdv',

'S2_2022-3_EVI_Stdv',
'S2_2022-3_red_Stdv',
'S2_2022-4_EVI_Stdv',
'S2_2022-4_MBI_Stdv',
'S2_2022-4_BSI_Mean',
'S2_2022-4_EVI_Mean',
'S2_2022-4_MBI_Mean',
'S2_2022-4_NDWI_Mean',
'S2_2022-4_RE4_Mean',
'S2_2022-4_SWIR2_Mean',
'S1_2021-1_VH_Mean',
'S1_2021-1_VV_Stdv',
'S1_2021-2_VH_Stdv',
'S1_2021-3_RVI_Stdv',
'S1_2021-6_RVI_Stdv',
'S1_2022-2_RVI_Stdv',
'S1_2022-4_RVI_Mean',
'S1_2022-6_VV_Mean',
'S1_2022-4_VH_Stdv',
'CS_DEM_Mean',
'CS_SLOPE_Mean',
'HAR_2022-1_sent_100_Stdv',
'HAR_2022-1_sent_100_Mean'