Land Cover Mapping Using Automated Machine Learning with Landsat Data

Jojene R. Santillan
Institute of Photogrammetry and GeoInformation, Leibniz University

Meriam Makinano-Santillan
Institute of Photogrammetry and GeoInformation, Leibniz University

https://doi.org/10.5109/7323384

出版情報：Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES). 10, pp.1021-1028, 2024-10-17. International Exchange and Innovation Conference on Engineering & Sciences
バージョン：

Land Cover Mapping Using Automated Machine Learning with Landsat Data

Jojene R. Santillan[1,2], Meriam Makinano-Santillan[2]
[1] Institute of Photogrammetry and GeoInformation, Leibniz University Hannover, Germany.
[2] Caraga Center for Geo-Informatics & Department of Geodetic Engineering, College of Engineering and Geosciences, Caraga State University, Butuan City, Philippines.
Corresponding author email: santillan@ipi-uni.hannover.de, {jrsantillan, mmsantillan}@carsu.edu.ph

Abstract: *This study explores the efficacy of AutoML in developing machine learning models for pixel-wise classification of land cover in Landsat images, focusing on Boracay Island, Philippines. Using the MLJar AutoML tool, high-performing algorithms including Neural Network, XGBoost, CatBoost, Extra Trees and LightGBM were integrated into an ensemble classifier through iterative selection and tuning processes. Evaluation on the training image demonstrated superior performance with strong precision and recall metrics across various land cover classes. However, variability in classifier performance was evident when applied to images from different dates or sensors, particularly affecting built-up areas and less prevalent classes. Despite this variability, significant land cover trends from 2008 to 2024 were discerned in Boracay Island, showing a substantial increase in built-up areas (23% to 38% of total area) and a decline in vegetation cover (59% to 45%). These findings underscore the dynamic changes occurring on the island and highlight the practical applications of this study, such as urban planning, environmental monitoring, and policymaking to balance development with environmental preservation, ensuring the island's long-term sustainability.*

Keywords: Automated Machine Learning; AutoML; Land Cover Classification; Urbanization; Green Spaces; Landsat; Boracay Island, Philippines.

1. INTRODUCTION

Urbanization is accelerating worldwide, transforming landscapes and ecosystems at an unprecedented rate. Built-up area development associated with the expansion of urban areas often results in the conversion of natural habitats, agricultural lands, and green spaces, leading to environmental degradation, altered microclimates, reduced biodiversity, and other negative impacts [1,2]. Monitoring these changes through land cover mapping is crucial for sustainable urban planning and environmental management. Satellite remote sensing technology plays an important role in this regard by facilitating the acquisition of multitemporal images, and their analysis for the creation of land cover maps to enhance monitoring capabilities [3].

Over the last few decades, the field of remote sensing has witnessed a paradigm shift with the adoption of machine learning (ML) and deep learning (DL) techniques for land cover mapping and change detection [4]. Unlike traditional methods that rely on manual interpretation of satellite imagery, machine learning algorithms automate the processes of pattern recognition, feature extraction, and classification. This approach not only improves efficiency but also enhances the accuracy and scalability of land cover analysis. However, the implementation of ML approach poses to be challenging. One of the primary challenges is the selection and configuration of appropriate models for specific applications. Users must choose from a variety of algorithms, such as Random Forests, Support Vector Machines (SVM), Artificial Neural Networks (ANNs), and others, each with its own strengths and limitations depending on the application, including the complexity and scale of the data [5,6]. Training and tuning an ML model is an iterative process and requires careful optimization of hyperparameters to maximize model performance in terms of accuracy, precision, recall, and other metrics relevant to the task at hand [7,8]. Given these challenges, proficiency in ML becomes imperative for users. However, in practice, such expertise is not universally attained.

Automated Machine Learning (AutoML) has emerged both as a powerful tool and a practical framework to address these challenges. AutoML automates several critical processes spanning from data preparation, feature engineering, model selection, training, optimization and evaluation to the application of the model, e.g., for regression or classification tasks [9]. The primary goal is to minimize the manual work associated with ML technologies, thereby speeding up their implementation. Hence, AutoML is beneficial as it enables professionals and non-experts alike to effectively utilize ML models "off-the-shelf," even with limited data science expertise [10].

Recently, the AutoML approach is starting to find applications in the remote sensing and geosciences fields. For example, AutoML has seen promising outcomes in susceptibility assessment for multiple forest disturbances [11], and in landslide susceptibility predictions [12]. A consensus from these cited examples is that not only AutoML simplified the process of ML model development and implementation, but it has also shown to have better accuracy, making it an attractive alternative to manual ML.

Despite its successful applications in diverse fields, AutoML remains relatively underexplored in the context of land cover classification, particularly in enabling timely and accurate monitoring of urban expansion, land use changes, and green space dynamics at both local and global scales. Hence, this study aims to assess the feasibility and effectiveness of AutoML in automatically classifying land cover types using Landsat satellite remote sensing data. Our overarching goal with the AutoML approach is to enhance the accuracy and efficiency of monitoring urban expansion using automated techniques and facilitate the detection and

analysis of temporal changes in urban and green space patterns. We evaluated our approach using Boracay Island, Philippines as a case study site, an area that has undergone substantial development over the past few decades. Specifically, we aim to address the following research questions:

- What ML models, or combinations thereof, can be developed using AutoML for the pixel-wise classification of land cover in Landsat images?
- How well does an AutoML-generated land cover classifier generalize when applied to Landsat images acquired on different dates or by other sensors compared to those used for model training?
- From the classified images, what land cover trends in the study area can be discerned and quantified considering the classifier's accuracy?

2. THE STUDY AREA

Boracay is a resort island situated in the Western Visayas region of the Philippines (Fig. 1). It lies about 0.8 kilometers off the northwest coast of Panay Island. Covering a total area of approximately 1,032 ha, it falls under the administration of three barangays within Malay, Aklan [13]. As of 2020, the island had a population of 37,802, consisting of residents, migrants, and stay-in workers, excluding tourists [14]. Boracay Island is famous for its stunning white sandy beaches and vibrant nightlife. This small island attracts millions of tourists annually, significantly contributing to both the local and national economies [13].

Over the last decades, Boracay has transformed from a largely subsistent agricultural community to a bustling hub of commercial and tourism activities [15]. This transformation has led to extensive urbanization, characterized by the expansion of built-up areas at the expense of green spaces. The increasing pressure from tourism-related infrastructure, such as hotels, restaurants, and recreational facilities, has significantly altered the island's landscape, including its surrounding waters [16]. The influx of tourists and the rapid and extensive development have raised critical environmental concerns, particularly regarding the sustainability of its natural resources and ecosystems. In 2018, starting on April 26, the Philippine government closed the island to local and foreign tourists for six months to address worsening ecological issues and environmental conditions [13]. The island reopened on 26 October 2018.

Despite rapid and extensive development on the island being a major cause of critical environmental concerns, there is limited comprehensive information on the dynamics of land cover changes on Boracay Island. Previous efforts, such as the land cover mapping exercise using ML in [17], primarily focused on comparing the accuracy of ML models and generating built-up/non-built-up maps without subsequent evaluation or analysis. The present study aims to address these gaps and contribute to a better understanding of the development of built-up areas and green space dynamics on the island.

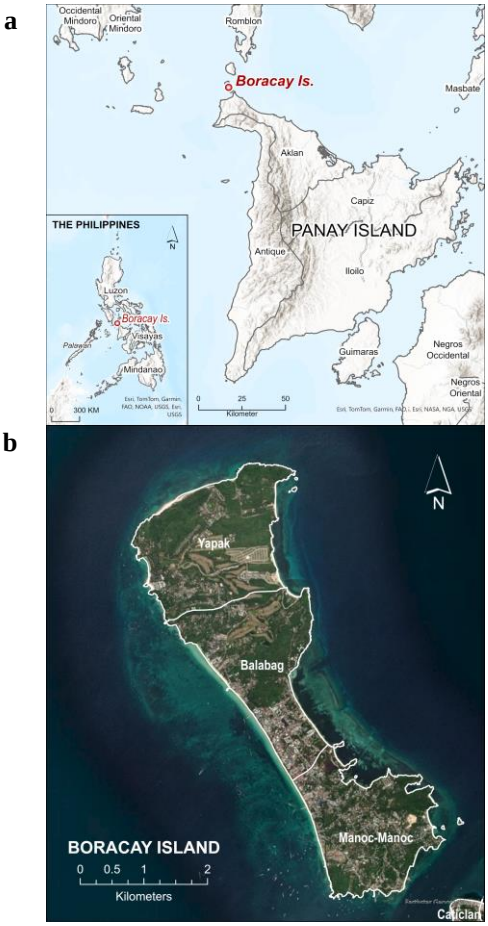


Fig. 1 **a** The geographical location of Boracay Island, in Malay, Aklan, Philippines. **b** A satellite view of Boracay from 2024, highlighting the boundaries of its three barangays. Satellite image credits: Maxar, Esri.

3. METHODOLOGY

3.1 Overview

This study utilizes the AutoML framework to develop a ML classifier for mapping land cover of Boracay Island using Landsat satellite images. Our classes of interests include built-up, vegetation, barren areas, sand, and water. We chose to utilize Landsat images primarily due to their long-term data availability, which will enable the classifier we develop to be used for multitemporal land cover mapping of the study area. Additionally, these images are available as science products, ensuring they are analysis-ready and geographically consistent across time scales. The methodology we employed can be summarized as follows. Firstly, we utilized a recently acquired (year 2024) Landsat 8 Operational Land Imager (OLI) image, along with labeled training data, as inputs to an AutoML tool to develop the classifier. To underscore the importance of AutoML, we employed a naïve approach where we relied on the AutoML tool for model selection and parameter optimization. The performance of the ML-based classifier developed in the previous step was then evaluated for accuracy. This was done by classifying an independent test dataset taken from the same image used in training the model, as well as by classifying images acquired on different dates or by different sensors [e.g., Landsat 5 Thematic Mapper (TM)]. For each image where the classifier was applied, validation was conducted to assess its generalization capability across various acquisition dates, ensuring its accuracy before utilization in land cover mapping. Using

the results of the accuracy assessment, we estimated the uncertainties in land cover classification and incorporated these uncertainties in land cover area estimation and in quantifying changes in land cover, particularly in built-up and vegetated areas of the island in recent years. The detailed methodology is discussed in the succeeding sections.

3.2 Landsat Images

We downloaded Landsat 5 TM and Landsat 8 OLI images of Boracay Island (Table 1) from the United States Geological Survey (USGS) Earth Explorer (https://earthexplorer.usgs.gov/). The acquisition dates for these images were chosen to ensure the availability of corresponding reference data for training and validation purposes (see Section 3.3). The images, in GeoTIFF format, belong to path 112, row 52 of the Landsat path/row World Reference System catalogue. They were delivered as Level 2 Science Products (L2SP), which means that the images have undergone radiometric calibration, terrain correction, and atmospheric correction. The downloaded Landsat 5 images consist of six surface reflectance (SR) bands (bands 1-5, 7) and one surface temperature (ST) band (band 6). Landsat 8 images have seven SR bands (bands 1-7) and one ST band (band 10). We opted to not utilize the ST band due to anomalous data pixels present in some of the images, hence, avoiding inconsistencies in the training data. Pixel values in both L2SP Landsat images are encoded as unsigned 16-bit integers, and with a spatial resolution of 30 meters. In all the downloaded images, the area where the island is located is cloud-free. Using a Geographic Information System (GIS) software (ArcGIS Pro 3.3), we extracted subsets containing only the island and its surrounding waters from each image for subsequent analysis. Each subset has dimensions of 171x233 pixels, totaling 39,843 pixels. Within these subsets, the main island occupies 11,642 pixels, which corresponds to approximately 1,047.78 ha. This is 15.78 ha more than the reported area of the island, mainly due to the coarse resolution of the Landsat images. In determining the main island boundary, we utilized the Philippine subnational administrative boundaries GIS Shapefile available at Humanitarian Data Exchange website (https://data.humdata.org/dataset/cod-ab-phl).

3.3 Training Data Preparation

Training data for ML classifier development was mainly collected from the 2024 May 18 Landsat 8 image. The dataset includes pixels manually labeled as built-up areas (class 1), barren areas (class 2), sand (class 3), vegetation (class 4), and water (class 5). The labeling was aided by high-resolution (pixel size < 1 m) satellite image acquired on 2024 June 6, which we obtained from Esri's World Imagery Wayback digital archive (https://livingatlas.arcgis.com/wayback/#active=39767&mapCenter=121.93212%2C11.97048%2C14).

To streamline the labeling process, a grid ("fishnet") matching the exact dimensions of the subset image pixels was created using ArcGIS Pro. A point vector file (Shapefile) was also generated corresponding to the centers of each pixel (i.e., each point corresponds exactly to one pixel). These vector layers (grid of square polygons and points) were then overlaid onto the high-resolution reference image for annotation purpose. A point is labeled as belonging to a particular class if at least 50% of the area within the square polygon is covered by that land cover. All identifiable pixels were labeled to maximize the quantity of training data available, a common requirement in ML model development. However, we noticed that barren areas and sand are underrepresented when focusing only on Boracay Island. To avoid class imbalance during model training, we opted to collect additional samples of these classes from nearby islands. From all the labeled points, we randomly selected 800 points for each class, resulting in a total of 4,000 points for model training. In addition to the class labels, we also extracted at each point the corresponding pixel values for each band of the Landsat image using the "Extract Multi Values to Points" tool in ArcGIS Pro. Only bands common to Landsat 5 and Landsat 8 OLI were utilized, as it our aim to test the generalization of the trained model when used to classify these types of images. These common bands are the blue, green, red, near infrared (NIR), short wave infrared 1 (SWIR1), and SWIR2. The final output of this step is a point Shapefile of the training data, along with its attribute table (*.dbf) containing the class labels and SR band values.

Table 1. List of Landsat images for machine learning classifier development and land cover classification, with their corresponding high-resolution satellite images for training and validation data collection.

Image Acquisition Date	Imaging Sensor	High-resolution Reference Image Acquisition Date
2008 March 19	Landsat 5 TM	2008 January 16
2020 August 27	Landsat 8 OLI	2020 May 20
2023 June 17	Landsat 8 OLI	2023 June 29
2024 May 2	Landsat 8 OLI	2024 June 6
2024 May 18	Landsat 8 OLI	2024 June 6

3.4 AutoML Using MLjar

We utilized MLjar (https://supervised.mljar.com/), an AutoML Python package aimed for building, training, and deploying ML models. Its core functionality includes automating fundamental tasks such as data preprocessing, constructing ML models, and performing hyperparameter tuning using systematic approaches like random search and hill climbing [18]. MLJAR leverages various ML libraries and frameworks, including *scikit-learn*, to implement its models. MLjar version 1.1.9 (https://github.com/mljar/mljar-supervised), which we used in this work, has the following ML models: Linear, Random Forest, Extra Trees, Light Gradient Boosting Machine (LightGBM), Xgboost, CatBoost, Neural Networks, and Nearest Neighbors. Due to space limitations, we do not provide detailed descriptions of these models in this paper. For further information, readers are encouraged to consult widely available references, including MLJar's website.

We used the MLjar AutoML app (https://github.com/mljar/automl-app) in classifier development. As MLJar has been designed to work with tabular data, we converted our training dataset into a comma space value (CSV) file, containing the class numbers, and band values. In the AutoML app, we selected the "Train AutoML (advanced)" mode. In addition to selecting the best algorithms, this mode also enables feature engineering (i.e, "golden features" generation), feature selection, stacking and ensembling algorithms, and cross-validation. "Golden features" are

those that have great predicted power; they are constructed based on original features (e.g., the Landsat bands in the training data), mainly using feature differences or ratios [19]. Stacking, including ensemble stacking, is an ML technique that combines multiple base models (learners) using a meta-model (or meta-learner) to make predictions. The idea behind stacking is to leverage the strengths of different models by learning how to best combine their predictions.

The classifier development process is straightforward. First, the CSV file containing the training data is uploaded to the application. The features for training are then selected, specifically the bands common to the three Landsat sensors: blue, green, red, NIR, SWIR1, and SWIR2. The class value (1-5) is set as the target for classification. We enabled advanced options such as golden features construction, feature selection, model stacking, and ensemble training. All available machine learning algorithms were considered. A 10-fold cross-validation strategy was employed, incorporating training data shuffling and stratification, with *log loss* (also called *cross-entropy loss*) as the evaluation metric. A time limit of 3000 seconds was set, which proved sufficient for the AutoML tool to perform the advanced options.

3.5 Classifier Performance Evaluation

The land cover classifier developed in the previous step has its own performance evaluation metrics reported after development. We conducted independent evaluation of the classifier's performance by testing it to image data not included in model training, including Landsat images of the study area acquired on different dates or by another sensor (TM). For this purpose, the point Shapefile (corresponding to the center of pixels within the study area) was used to extract SR band values from all the Landsat images, and individual CSV files were generated for each image (Table 1). Each CSV file was then uploaded to the AutoML app for class prediction by the trained classifier.

Each classification output was subjected to accuracy assessment. We employed good practice recommendations and workflow [20] to ensure unbiased estimation of the classification accuracies, including quantification of uncertainties due to misclassifications in land cover area estimation. This includes stratified random sampling of validation (test) samples from the classified images. As the study area is relatively small, the total number of test samples (n) was estimated using the following sample size formula [20]:

$$n = \frac{(\sum W_i S_i)^2}{\left[S(\hat{O})\right]^2 + \frac{\sum W_i S_i^2}{N}} \quad (1)$$

where N = number of pixels in the study area (i.e., 11,642 within the mainland Boracay), $S(\hat{O})$ represents the standard error of the estimated overall accuracy we aim to achieve (set to 0.01), W_i is the mapped ("classified") proportion of class i, and S_i is the class i standard deviation; $S_i = \sqrt{U_i(1-U_i)}$, where U_i is the conjectured user's accuracy of class i. The user's accuracies are included in the trained model performance report. However, we decided to not use the calculated values as the actual user's accuracies when applied to the other image data could be well below the reported trained model accuracies, especially that the model was only trained using a specific image data. For purposes of test sample size calculations, we set U_i to 0.50 (i.e., we conjectured that the classifier's user's accuracy for all classes is 50%).

The number of stratified random samples per image and per class are listed in Table 2. For the image used in model development, only those pixels not included in the training data were randomly selected per class. Each test sample was labeled using the corresponding high-resolution satellite images obtained from Esri's World Imagery Wayback digital archive and from the Google Earth Pro application. Afterwards, error matrices were generated per classified image, showing the distribution of correctly and incorrectly classified pixels across each class, which helps in assessing the classification accuracy and identifying the specific areas where the model performs well or needs improvement. Various metrics (Table 3) where utilized in the accuracy assessment and model performance evaluation. *Overall Accuracy* measures the proportion of correctly classified pixels out of the total pixels. *Producer's Accuracy (Recall)* indicates the ability of the classifier to correctly label all pixels of a particular class, reflecting how well the actual land cover types are represented in the classification. *User's Accuracy (Precision)* assesses the reliability of the classification by measuring the proportion of pixels classified as a certain class that are actually that class. The *F1 Score* is a harmonic mean of precision and recall, offering a single metric to assess the balance between correctly identifying positive instances (precision) and capturing all positive instances (recall) in a classification task.

3.6 Land Cover Area Estimation

Land cover classification aims to accurately determine the extent of specific land cover types. Typically, the area of each class is derived directly from the land cover map generated by the classifier. However, classification errors introduce biases in these maps [21]. Commission errors occur when the classifier incorrectly assigns a pixel to a class, leading to an overestimation of the area for that class. Conversely, omission errors occur when the classifier fails to correctly identify a pixel belonging to a class, resulting in an underestimation of the area for that class. To overcome these limitations, we adopted a procedure described in [20,21] for unbiased area estimation. The steps include converting the conventional error matrix into an error matrix of estimated area proportions. This new error matrix together with the mapped proportions of land cover classes are used as inputs to a stratified, unbiased area estimator, including its 95% confidence interval. The estimated area for each class is considered "error-adjusted" as it includes area of omission error and excludes area corresponding to the commission error [21].

4. RESULTS AND DISCUSSION

4.1 AutoML-generated Classifier

Fig. 2 presents the primary results of the classifier development using the MLJar AutoML tool. Specifically, Fig. 2a illustrates the models that were iteratively selected and tuned, alongside their corresponding log loss values at each iteration. A lower log loss value indicates superior model performance. It can be noted that Linear, Random Forest, Extra Trees, and Nearest Neighbors

exhibited higher log loss values with greater variability, suggesting suboptimal performance relative to other models. In contrast, algorithms like XGBoost, CatBoost, Neural Network, and LightGBM consistently demonstrated lower log loss values, indicative of enhanced performance in the land cover classification task. This trend is further exemplified in Fig. 2b.

It is important to note that each point in Fig. 2a represents a trained ML algorithm or model. The iterative process enabled the AutoML tool to identify and prioritize algorithms that consistently performed better throughout the iterations. Consequently, these high-performing algorithms were combined into an ensemble. Fig. 2c depicts the architecture of the optimal classifier, designated as "Ensembled_Stacked." This classifier is composed of individually trained algorithms, stacked algorithms (the same algorithm trained in different ways), and a "sub-ensemble" that itself comprises several individually trained algorithms. Each component within the ensemble contributes to the final prediction according to its assigned weight. The weights represent the importance or reliability of each model's prediction, as determined during the training process.

Looking closely at Fig. 2c, we can see that the main ensemble and the sub-ensemble are dominated by the Neural Network algorithm, and with few instances of Xgboost, CatBoost, LightGBM, ExtraTrees, and Nearest Neighbors. The feature engineering and feature selection capabilities of AutoML is evident, with some of the component algorithms utilizing Golden Features and KMeans Features. The golden features that were generated and utilized include the sum, difference, and ratio of Landsat bands. Trained algorithms utilizing KMeans Features means that the AutoML tool has utilized the KMeans clustering algorithm to create additional features from the training dataset.

The developed classifier has class precisions, class recalls, class F1 scores, and overall accuracy greater than 0.90 (Fig. 2d), indicating more than satisfactory classification performance. The classifier has better precision and recall in classifying all classes, although such characteristics is relatively lower in classes 1 and 3 (built-up areas, sand).

Table 2. Number of test samples for independent evaluation of the AutoML-generated land cover classifier. Class designations: 1- built-up areas, 2 – barren areas, 3 – sand, 4 – vegetated areas, 5 – water.

Classified Image Date	Classified as					
	1	2	3	4	5	Total
2008 March 19	192	74	67	500	16	849
2020 August 27	261	52	56	439	11	819
2023 June 17	245	52	47	522	20	886
2024 May 2	275	82	42	306	10	715
2024 May 18	262	79	43	325	12	721

Table 3. Classification accuracy and model performance evaluation metrics. Notations: *TP* – true positives, *FP* – false positives, *FN* – false negatives; *n* = total number of test samples.

Metric	Formula
User's Accuracy (Precision)	$\frac{TP}{TP + FP}$
Producer's Accuracy (Recall)	$\frac{TP}{TP + FN}$
F1 Score	$2 \times \frac{Precision \times Recall}{Precision + Recall}$
Overall Accuracy	$\frac{TP}{n}$

4.2 Independent Evaluation of Classifier Performance

Fig. 3 shows the land cover maps that were generated by applying the AutoML-generated classifier to Landsat images. The results of the accuracy assessment of the land cover maps using an independent set of validation dataset are summarized in Tables 4 to 8. In our accuracy assessment, the water and sand classes were combined with the barren areas due to their minimal distribution and re-labeled as "Others." This merging allows us to focus on the primary land cover classes of interest in this study: built-up and vegetation.

The independent validation of the classification result from 2024 May 18 image (Table 4) shows relatively consistent accuracy with the accuracy reported after model development (Figure 2e), particularly for the built-up and vegetation classes. The overall accuracy based on the validation is high at 0.953 (or 95.3%). The classifier maintained or even surpassed its values of precision, recall, and F1 scores for the built-up and vegetation classes. However, for other classes, the precision decreased to 0.858, indicating that the classifier incorrectly classified some built-up or vegetated areas as belonging to these other classes.

When the classifier was applied to the 2024 May 2 Landsat 8 OLI image, the validation results showed a considerable decrease in overall accuracy and most class-based metrics (Table 5). The overall accuracy dropped from 0.953 (based on the 18 May 2024 image) to 0.866. Additionally, the precision for the built-up area class was low at 0.738, indicating that the classifier overestimated built-up areas in this image. However, the recall for the built-up area class remained high at 0.958, showing that the classifier correctly identified the majority of built-up area pixels. In contrast, for the vegetation class, the precision was high at 0.993, but the recall was lower at 0.835, indicating that the classifier underestimated vegetated areas in this image.

The drop in overall accuracy is also evident in the validation results for the other classified images (Tables 6-8). The overall accuracies were 0.894 for the 2023 June 17 Landsat 8 OLI image, 0.879 for the 2020 August 27 Landsat 8 OLI image, and 0.881 for the 2008 March 19, Landsat TM image. While the precision for the built-up area class also decreased, it remained above 0.80. The recall values were above 0.80, except for the 2008 result, where it was 0.795. Notably, the precision and recall values for the vegetation class consistently remained high (above 0.90), indicating that the classifier performed satisfactorily in classifying this land cover class. For the other classes, the accuracy metrics also decreased and

showed consistency with the results for the other images. These findings suggest that the generalization capability of the AutoML-generated classifier, when applied to images acquired on different dates or by different sensors, demonstrates variability in performance. The classifier consistently performed well in identifying vegetation, with high precision and recall values across all images, indicating a strong generalization capability for this class. The classifier's performance for built-up areas showed variability, with precision dropping to as low as 0.738 in one instance, indicating an overestimation issue. For other classes, the precision and recall metrics were lower, particularly when classes were merged or minimal in distribution, suggesting the classifier struggles to generalize effectively for these less prevalent classes. Moreover, the notable drop in overall accuracy when the classifier was applied to images from different dates or sensors highlights the challenge of maintaining high performance across varied datasets.

4.3 Land Cover Trends and Area Estimates

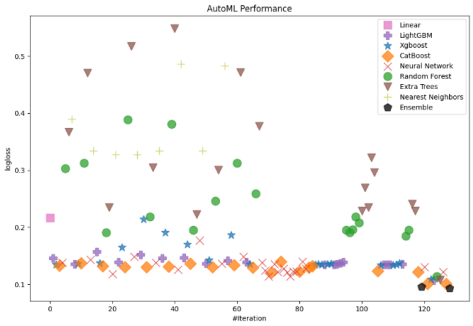
Despite a decrease in accuracy when the classifier was applied to images from different dates and sensors, it maintained a moderate level of accuracy, allowing for the analysis of land cover trends in the study area. Additionally, detailed class-specific accuracies for each classified image facilitated the estimation of land cover class areas, including their 95% confidence intervals (Table 9).

The generated land cover maps (Fig. 3) reveal a significant increase in built-up area development on the island from 2008 to 2024. In 2008, the estimated built-up area was 240.72 ± 19.77 ha, comprising approximately 23% of the island's total area. By 2024, this area had increased by approximately 64% to 394.88 ± 14.45 ha, equivalent to an average annual increase of about 9.64 ha over a 15-year period. In 2024, built-up areas now cover about 38% of the island's total area. Most built-up areas are concentrated in the central and southern parts of the island, with development extending towards the north. The northern part of the island consistently exhibits notable barren areas.

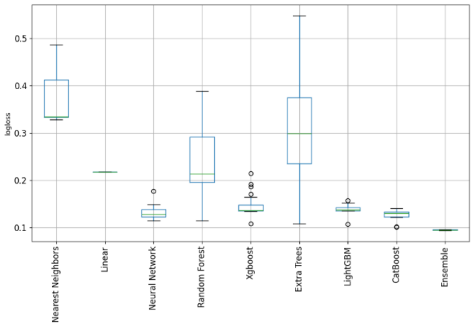
Vegetation cover was predominant from 2008 to 2023, but its dominance has declined over time. In 2008, the island had 620.9 ± 17.91 ha of vegetation, which accounted for approximately 59% of the island's total land area and remained relatively stable until 2023. By 2024, vegetation cover had decreased to 469.24 ± 8.96 ha, representing around 45% of the total area. This reduction in vegetation cover amounts to approximately 9.48 ha per year, which is roughly equivalent to the annual increase observed in built-up areas on the island.

Considering the uncertainties associated with the land cover classifications, the detected changes in both built-up and vegetation areas appear realistic and are not solely due to misclassifications. The substantial increase in built-up areas and the corresponding decrease in vegetation cover are greater than the uncertainties indicated by the 95% confidence intervals, suggesting that these observed changes are likely genuine. While some variability and misclassification may be present, the confidence intervals for these measurements indicate a relatively high level of precision, supporting the validity of the trends. The parallel rates of increase in built-up areas and decrease in vegetation cover align with expected patterns of urban development in popular tourist destinations like Boracay Island, reinforcing the likelihood that these trends reflect real changes on the island, despite potential classification challenges.

a



b



c

Main ensemble ("Stacked")

Model	Weight
104_Xgboost_Stacked	8
35_CatBoost_GoldenFeatures_Stacked	3
50_ExtraTrees_Stacked	19
55_NeuralNetwork	7
57_NeuralNetwork	1
58_NeuralNetwork_GoldenFeatures	15
58_NeuralNetwork_KMeansFeatures	10
61_NeuralNetwork	2
69_NearestNeighbors	2
71_NeuralNetwork_GoldenFeatures	20
73_NeuralNetwork	2
74_CatBoost_GoldenFeatures_Stacked	11
75_NeuralNetwork	2
76_NeuralNetwork	3
85_LightGBM_Stacked	10
Ensemble	12

Sub-ensemble

Model	Weight
10_Xgboost	9
55_NeuralNetwork	8
57_NeuralNetwork	1
58_NeuralNetwork_GoldenFeatures	20
58_NeuralNetwork_KMeansFeatures	16
5_Default_NeuralNetwork	1
61_NeuralNetwork	3
69_NearestNeighbors	2
71_NeuralNetwork_GoldenFeatures	35
73_NeuralNetwork	4
75_NeuralNetwork	2
76_NeuralNetwork	7

d

	1	2	3	4	5	accuracy
precision	0.924877	0.980025	0.957854	0.992481	0.987593	0.9685
recall	0.93875	0.98125	0.9375	0.99	0.995	0.9685
f1-score	0.931762	0.980637	0.947568	0.991239	0.991283	0.9685

Fig. 2 Results of classifier development using MLJar AutoML. **a**. Leaderboard showing the performance of various ML models. **b**. Boxplots of model log loss. **c** The optimal classifier ("best model"): "Ensembled_Stacked" (models + sub-ensemble of models), shown here with its structure. **d** Error matrix. **e**. Accuracy metrics of the optimal classifier.

5. CONCLUSIONS AND OUTLOOK

This study makes significant contributions in two key areas: (1) the use of AutoML for land cover classification, and (2) quantifying recent land cover changes on Boracay Island. To the best of our knowledge, this is the first study that demonstrated the efficacy of utilizing AutoML for developing ML models tailored to pixel-wise classification of land cover in Landsat images. Through iterative selection and tuning processes, the MLJar AutoML tool effectively identified and integrated high-performing algorithms such as XGBoost, CatBoost, Neural Network, LightGBM, and Extra Trees into an ensemble classifier. While it showed superior performance with high precision and recall on the training image, its accuracy varied with images from

different dates or sensor, particularly struggling with built-up areas and less prevalent classes.

Despite variability in classifier accuracy, the land cover maps reveal significant trends: built-up areas increased from 23% to 38% of the island, while vegetation cover declined from 59% to 45% between 2008 and 2024.

This study has practical applications for urban planning, environmental monitoring, and policymaking on Boracay Island. Urban planners can use these trends to guide sustainable development and land use decisions. Environmental agencies can make use of the tracked changes in built-up and vegetation areas to monitor ecological impacts and implement conservation measures. Policymakers can leverage the land cover maps to create regulations that balance development with environmental preservation, ensuring the island's long-term sustainability. Finally, the tourism industry can use the findings as an important input in managing growth in a way that protects natural resources and maintains the island's appeal.

Further work involves refining the AutoML-derived classifier, particularly by incorporating additional training data from various dates and sensors. This will enhance the classifier's robustness and generalization across different temporal and sensor-based datasets.

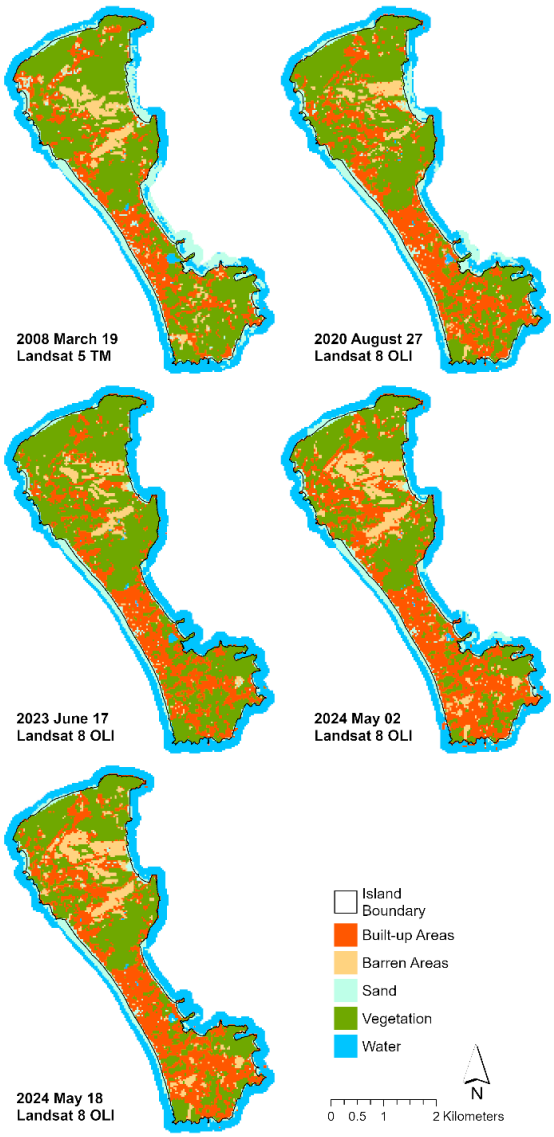


Fig. 3 Land-cover maps of Boracay Island generated by applying the AutoML-generated classifier to Landsat images.

Table 4. Summary of results from the independent validation of the AutoML-generated classifier: **2024 May 18 Landsat 8 OLI** image

Accuracy Metric	Built-up	Vegetation	Others	Mean
Precision	0.966	0.982	0.858	0.935
Recall	0.930	0.988	0.913	0.943
F1 Score	0.948	0.985	0.885	0.939
Overall Accuracy	0.953			

Table 5. Summary of results from the independent validation of the AutoML-generated classifier: **2024 May 2 Landsat 8 OLI** image

Accuracy Metric	Built-up	Vegetation	Others	Mean
Precision	0.738	0.993	0.836	0.856
Recall	0.958	0.835	0.806	0.866
F1 Score	0.834	0.907	0.821	0.854
Overall Accuracy	0.866			

Table 6. Summary of results from the independent validation of the AutoML-generated classifier: **2023 June 17 Landsat 8 OLI** image

Accuracy Metric	Built-up	Vegetation	Others	Mean
Precision	0.882	0.929	0.765	0.858
Recall	0.864	0.929	0.798	0.864
F1 Score	0.873	0.929	0.765	0.861
Overall Accuracy	0.894			

Table 7. Summary of results from the independent validation of the AutoML-generated classifier: **2020 August 27 Landsat 8 OLI** image

Accuracy Metric	Built-up	Vegetation	Others	Mean
Precision	0.812	0.934	0.824	0.857
Recall	0.906	0.919	0.705	0.843
F1 Score	0.857	0.927	0.824	0.848
Overall Accuracy	0.879			

Table 8. Summary of results from the independent validation of the AutoML-generated classifier: **2008 March 19 Landsat 5 TM** image.

Accuracy Metric	Built-up	Vegetation	Others	Mean
Precision	0.807	0.944	0.771	0.841
Recall	0.795	0.938	0.801	0.845
F1 Score	0.801	0.941	0.786	0.843
Overall Accuracy	0.881			

Table 9. Estimates of land cover area with a 95% confidence interval for Boracay Island based on Landsat image classifications. All values are reported in ha.

Classified Image Date	Built-up	Vegetation	Others
2008 March 19	240.72 ± 19.77	620.9 ± 17.91	186.17 ± 17.99
2020 August 27	299.49 ± 19.47	570.57 ± 19.37	177.72 ± 18.77
2023 June 17	295.18 ± 17.64	617.52 ± 18.93	135.09 ± 15.34
2024 May 18	394.88 ± 14.45	469.24 ± 8.96	183.66 ± 14.89

ACKNOWLEDGEMENT
The lead author thanks the Philippines' Department of Science and Technology - Science Education Institute (DOST-SEI) Foreign Graduate Scholarships in Priority S&T Fields and Caraga State University, Philippines, for the doctoral scholarship and fellowship that made this study possible.

REFERENCES

[1] B. Güneralp, M. Reba, B. U. Hales, E. A. Wentz, and K. C. Seto, "Trends in urban land expansion, density, and land transitions from 1970 to 2010: A global synthesis," Environmental Research Letters, vol. 15, no. 4, 2020, doi: 10.1088/1748-9326/ab6669.

[2] I. M. S. Abouelhamd, R. N. Onkangi, P. van der Kuil, and U. J. Walthe, "Investigating the Influence of Urban Density and Concentrations on Commuting Distance and Time: Empirical Evidence from Suhag City," Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES), vol. 9, pp. 279–286, Oct. 2023, doi: 10.5109/7157985.

[3] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images: From pixel-based to object-based approaches," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 80, pp. 91–106, 2013, doi: 10.1016/j.isprsjprs.2013.03.006.

[4] A. Vali, S. Comai, and M. Matteucci, "Deep learning for land use and land cover classification based on hyperspectral and multispectral earth observation data: A review," Remote Sens (Basel), vol. 12, no. 15, 2020, doi: 10.3390/RS12152495.

[5] A. Patil and S. Panhalkar, "A comparative analysis of machine learning algorithms for land use and land cover classification using google earth engine platform," Journal of Geomatics, vol. 17, no. 2, pp. 226–233, 2023, doi: 10.58825/jog.2023.17.2.96.

[6] G. S. B. Raju, C. Manasa, N. D. Bhavani, J. Amulya, and D. Shirisha, "Comparative Analysis of Different Machine Learning Algorithms on Different Datasets," in Proceedings of the 7th International Conference on Intelligent Computing and Control Systems, ICICCS 2023, 2023, pp. 104–109. doi: 10.1109/ICICCS56967.2023.10142906.

[7] S. Georganos et al., "Less is more: optimizing classification performance through feature selection in a very-high-resolution remote sensing object-based urban application," GIsci Remote Sens, vol. 55, no. 2, pp. 221–242, 2018, doi: 10.1080/15481603.2017.1408892.

[8] J. Galupino and J. Dungca, "Estimation of Permeability of Soil-Fly Ash Mix using Machine Learning Algorithms," Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES), vol. 9, pp. 28–33, Oct. 2023, doi: 10.5109/7157938.

[9] S. K. K. Santu, M. M. Hassan, M. J. Smith, L. Xu, C. Zhai, and K. Veeramachaneni, "AutoML to Date and Beyond: Challenges and Opportunities," ACM Comput Surv, vol. 54, no. 8, 2022, doi: 10.1145/3470918.

[10] J. Waring, C. Lindvall, and R. Umeton, "Automated machine learning: Review of the state-of-the-art and opportunities for healthcare," Artif Intell Med, vol. 104, 2020, doi: 10.1016/j.artmed.2020.101822.

[11] R. Eker, K. C. Alkiş, and A. Aydın, "Assessment of large-scale multiple forest disturbance susceptibilities with AutoML framework: an Izmir Regional Forest Directorate case," J For Res (Harbin), vol. 35, no. 1, 2024, doi: 10.1007/s11676-024-01723-9.

[12] G. Tang, Z. Fang, and Y. Wang, "Global landslide susceptibility prediction based on the automated machine learning (AutoML) framework," Geocarto Int, vol. 38, no. 1, 2023, doi: 10.1080/10106049.2023.2236576.

[13] C. M. Reyes, J. R. G. Albert, F. M. A. Quimba, Ma. K. P. Ortiz, and R. D. Asis, "The Boracay Closure: Socioeconomic Consequences and Resilience Management," Dec. 2018, Philippine Institute for Development Studies, Quezon City. Accessed: Jul. 04, 2024. [Online]. Available: https://pidswebs.pids.gov.ph/CDN/PUBLICATIONS/pidsdps1837.pdf

[14] Philippine Statistics Authority, "2020 Census of Population and Housing (2020 CPH) Population Counts Declared Official by the President." Accessed: Jul. 04, 2024. [Online]. Available: https://psa.gov.ph/content/2020-census-population-and-housing-2020-cph-population-counts-declared-official-president

[15] W. J. Trousdale, "Governance in context : Boracay Island, Philippines," Ann Tour Res, vol. 26, no. 4, pp. 840–867, 1999, doi: 10.1016/S0160-7383(99)00036-5.

[16] V. G. Limates, V. C. Cuevas, and E. Benigno, "Water quality and nutrient loading in the coastal waters of Boracay Island, Malay, Aklan, central Philippines," Journal of Environmental Science and Management, vol. 2016, no. Special Is, pp. 15–29, 2016.

[17] W. W. N. P. Akeboshi et al., "Land Cover Mapping of Boracay Island, Philippines Using Remote Sensing and Machine Learning," in 2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management, HNICEM 2022, 2022. doi: 10.1109/HNICEM57413.2022.10109464.

[18] Płońska A. and Płoński P., "MLJAR: State-of-the-art Automated Machine Learning Framework for Tabular Data. Version 1.1.9," 2024, MLJAR, Poland. [Online]. Available: https://github.com/mljar/mljar-supervised

[19] MLJAR Inc., "Golden Features," AutoML mljar-supervised. Accessed: Jul. 15, 2024. [Online]. Available: https://supervised.mljar.com/features/golden_features/

[20] P. Olofsson, G. M. Foody, M. Herold, S. V. Stehman, C. E. Woodcock, and M. A. Wulder, "Good practices for estimating area and assessing accuracy of land change," Remote Sens Environ, vol. 148, pp. 42–57, May 2014, doi: 10.1016/j.rse.2014.02.015.

[21] P. Olofsson, G. M. Foody, S. V. Stehman, and C. E. Woodcock, "Making better use of accuracy data in land change studies: Estimating accuracy and area and quantifying uncertainty using stratified estimation," Remote Sens Environ, vol. 129, pp. 122–131, Feb. 2013, doi: 10.1016/j.rse.2012.10.031.