INDIAN INSTITUTE OF TECHNOLOGY KANPUR
Department of Civil Engineering
2024-25

CE672: Machine Processing of Remotely Sensed Data

DIFFERENT TYPES OF ACCURACY ANALYSIS PROCEDURES

Presented By:
- Ayush Dixit (220262)
- Ashwani Kumar Singh (220244)
- Shreya Gupta (221027)
- Saloni Mittal (220942)

Submitted To:
Prof. Onkar Dikshit

TABLE OF CONTENTS

ABSTRACT

This paper analyzes the classification accuracy of remotely sensed images using statistical validation techniques. It emphasizes the use of confusion matrices to assess overall, user's, and producer's accuracy, while also introducing advanced indices like Kappa and Tau that adjust for chance agreement. Statistical tests and conditional metrics are discussed for comparative evaluation, highlighting the strengths and limitations of each method and the importance of selecting suitable accuracy measures for reliable map validation.

INTRODUCTION

01 Background

Satellite image classification is vital in remote sensing for applications like land cover mapping and environmental monitoring. Its reliability depends on robust accuracy assessment methods, which have evolved from simple comparisons to matrix-based evaluations. The confusion matrix remains a central tool, with growing emphasis on assessing accuracy from the end-user's perspective as applications diversify.

02 Significance of the Study

A classification result is only as valuable as its accuracy. While overall accuracy is commonly used, it can overlook class-specific errors. Complementary metrics like the Kappa coefficient, user's accuracy, and producer's accuracy offer a more complete picture. This study compares these metrics to guide practitioners in choosing the most suitable tools for their needs.

03 Objectives

The main objectives of this study are to implement and understand various accuracy assessment techniques for image classification, compare the effectiveness of metrics like overall, user's, and producer's accuracy along with the Kappa coefficient, and examine the impact of sampling methods and ground truth data on accuracy outcomes. These objectives follow standard remote sensing methodologies.

LITERATURE SURVEY

Foundational Work

- **Russell G. Congalton (1991):** Established the confusion matrix as the standard for accuracy assessment in remote sensing. Warned against overreliance on overall accuracy.
- **Giles M. Foody (2002):** Advocated using multiple accuracy metrics (User's, Producer's, Kappa) for reliable interpretation, especially with imbalanced classes.
- **Pontus Olofsson et al. (2014):** Proposed best practices—stratified random sampling, area estimation, and inclusion of confidence intervals for rigorous validation.

Recent Advances

- **Cheng Zhang et al. (2020):** Used Sentinel-2 imagery and spectral indices to show strong performance of pixel-based classifiers like Random Forest and KNN.
- **Google & World Resources Institute (2021):** Developed the Dynamic World V1 dataset using deep learning on Sentinel-2, offering daily, 10m land cover classification.

Gaps in Literature

- **Temporal data underused:** Most studies, including this one, use single-date imagery (e.g., April 2021); time-series can reveal dynamic land changes.
- **Lack of regional validation:** Global models often lack local testing, limiting their accuracy in Indian contexts.
- **Limited use of deep learning:** Classical classifiers like KNN are simple, but CNNs/U-Nets offer higher accuracy and spatial detail.
- **Minimal ground verification:** Many studies lack on-ground validation, reducing trust in classification outcomes.

ACCURACY ASSESSMENT METHODS

Confusion Matrix

- Standard tool for evaluating classification accuracy.
- Summarizes true vs. predicted classes.
- Provides overall, user’s, and producer’s accuracy.
- Captures errors of omission and commission.

Naïve Accuracy Measures

- Includes: Overall Accuracy (Ao), User’s Accuracy (Ci), Producer’s Accuracy (Oj).
- Simple and intuitive.
- Derived directly from the confusion matrix.
- Does not account for chance agreement.

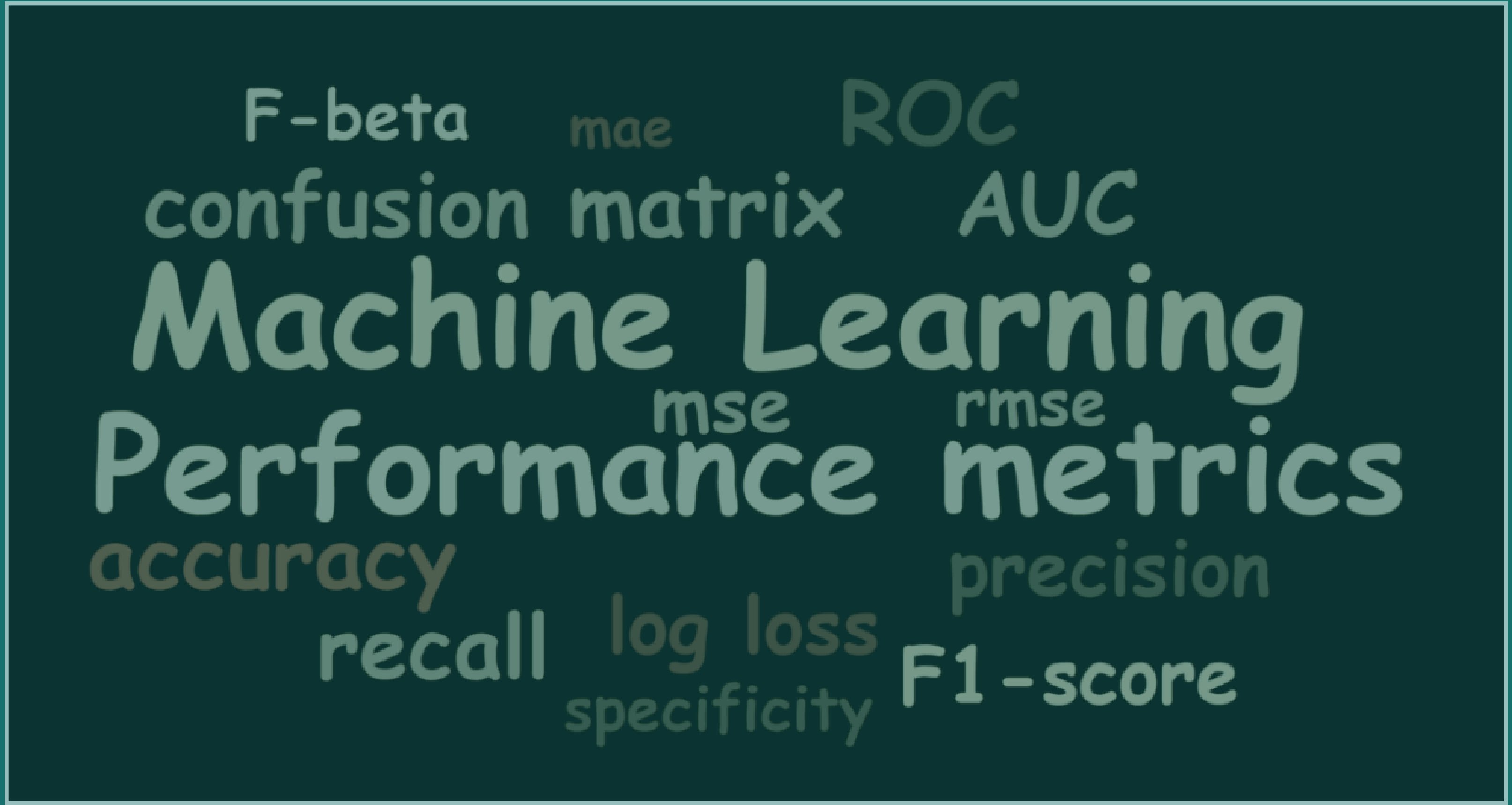
Tau Coefficient (τ)

- Introduced as an alternative to Kappa.
- Works better with unequal or unknown class priors.
- Directly interpretable and statistically testable.
- Useful in modern, real-world remote sensing tasks.

Kappa Coefficient (κ)

- Adjusts accuracy for chance agreement.
- Widely used in remote sensing.
- More statistically robust than naive metrics.
- Formula uses observed and expected agreement.

THEORY



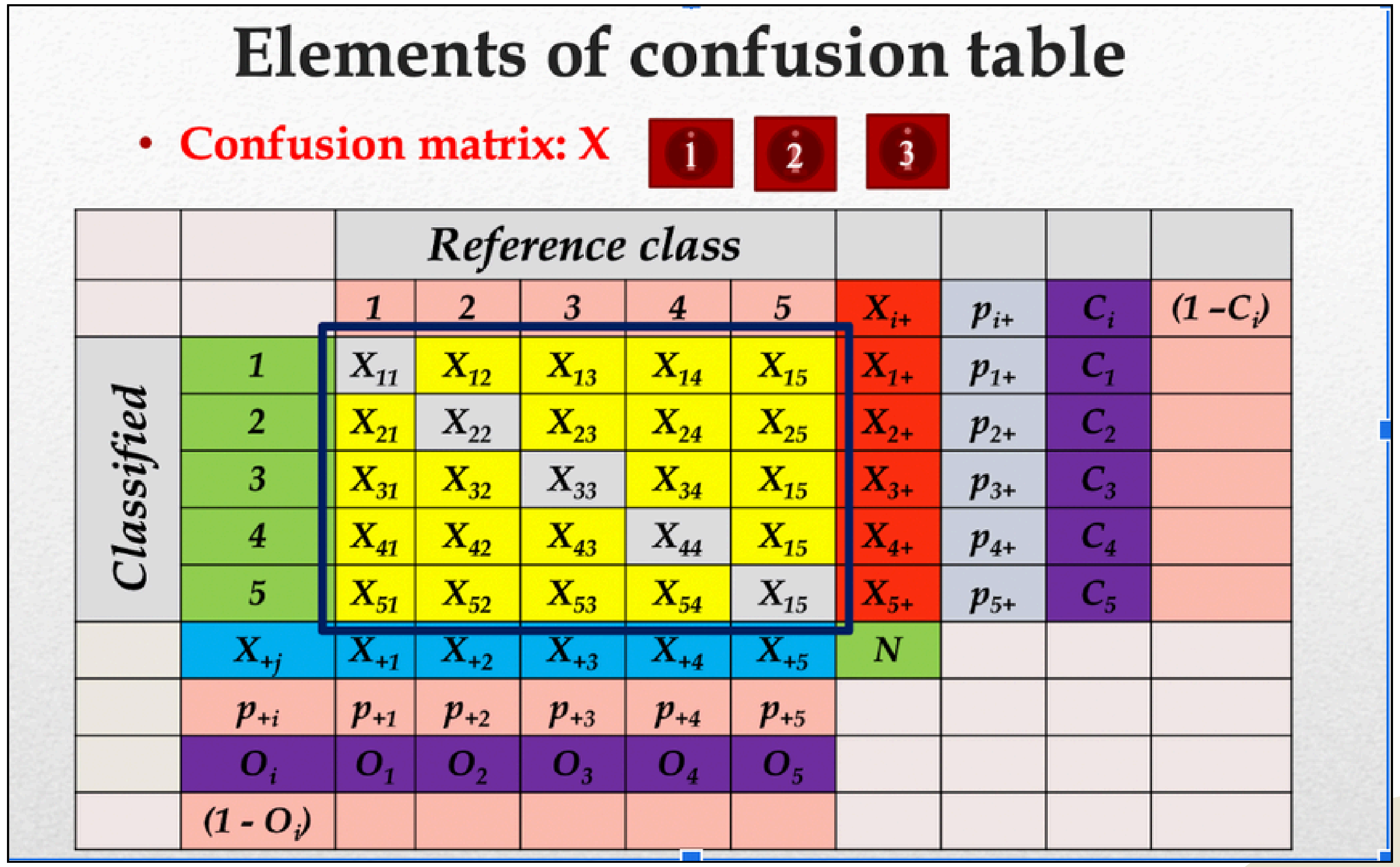
In remote sensing, classification of satellite images into thematic maps is followed by accuracy assessment to ensure the results reflect ground reality. Statistical evaluation using ground-truth data is the standard approach. Tools like the confusion matrix and Kappa coefficient quantify how closely the classified data matches reference data, offering insights into classification quality.

Confusion Matrix (Error Matrix)

The confusion matrix is an M × M table where:

- Rows represent the classified data.
- Columns represent the reference (true) data.
- Diagonal elements (X_{ii}) represent correct classifications.

Off-diagonal elements (X_{ij}, $i \neq j$) show misclassifications.



Elements of confusion table

- Confusion matrix: X 1 2 3

		Reference class								
		1	2	3	4	5	X_{i+}	p_{i+}	C_i	$(1-C_i)$
Classified	1	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{1+}	p_{1+}	C_1	
	2	X_{21}	X_{22}	X_{23}	X_{24}	X_{25}	X_{2+}	p_{2+}	C_2	
	3	X_{31}	X_{32}	X_{33}	X_{34}	X_{15}	X_{3+}	p_{3+}	C_3	
	4	X_{41}	X_{42}	X_{43}	X_{44}	X_{15}	X_{4+}	p_{4+}	C_4	
	5	X_{51}	X_{52}	X_{53}	X_{54}	X_{15}	X_{5+}	p_{5+}	C_5	
	X_{+j}	X_{+1}	X_{+2}	X_{+3}	X_{+4}	X_{+5}	N			
	p_{+i}	p_{+1}	p_{+2}	p_{+3}	p_{+4}	p_{+5}				
	O_i	O_1	O_2	O_3	O_4	O_5				
	$(1 - O_i)$									

- Error of Commission ($1 - C_i$): A class is wrongly assigned (false positive).
- Error of Omission ($1 - O_i$): A class is missed where it should be assigned (false negative).

THEORY

Tau Coefficient (τ)

$$\tau = \frac{p_o - \theta}{1 - \theta}$$

Where:

- p_o = observed agreement (same as in Kappa),
- θ = chance agreement based on prior probabilities of class membership

Weighted Measures

In real-world applications, not all misclassifications are equally severe. To account for varying degrees of error, a weight matrix W is introduced, where:

- w_i = 1: indicates perfect agreement.
- $w_{ij} \in [0,1]$: represents the penalty for misclassifying class i as class j, based on the severity of error.
- Using this matrix, weighted accuracy and weighted Kappa (κ_w) can be computed to reflect the relative seriousness of misclassifications.

Comparison of Classifications

$$Z = \frac{\hat{p}_a - \hat{p}_b}{\sqrt{s_a^2 + s_b^2}}$$

- $p\hat{}_a$, $p\hat{}_b$ are the accuracies of the two classification maps.
- s^2_a, s^2_b are their respective variance estimates.

Naïve Accuracy Measures

These are direct estimates from the confusion matrix without accounting for chance agreement.

1. Overall Accuracy (A_o)

$$A_o = \frac{\sum X_{ii}}{N}$$

2. User's Accuracy (C_i)

$$C_i = \frac{X_{ii}}{X_{i+}}$$

3. Producer's Accuracy (O_j)

$$O_j = \frac{X_{jj}}{X_{+j}}$$

4. Standard Errors and Confidence Intervals

$$\hat{s} = \sqrt{\frac{\hat{p}(1-\hat{p})}{N}} \qquad \hat{p} \pm \left[Z_{1-\alpha} \cdot \hat{s} + \frac{1}{2N}\right]$$

Where:

- X_{ii} are the diagonal elements of the confusion matrix representing correct classifications for class *i*,
- N is the total number of classified samples.
- X_{i+} is the total number of samples classified as class *i* by the model (row total).
- X_{+j} is the total number of reference samples belonging to class *j* (column total).
- Z_{1-a} is the standard normal deviate corresponding to the confidence level (e.g., 1.96 for 95%),
- 1/(2N) is a finite sample correction factor.
- $p\hat{}=A_o$

Kappa Coefficient (κ)

Kappa improves upon overall accuracy by adjusting for agreement that may happen by chance. It is defined as:

$$\kappa = \frac{p_o - p_c}{1 - p_c} \qquad p_c = \sum (p_{i+} \cdot p_{+i})$$

Where:

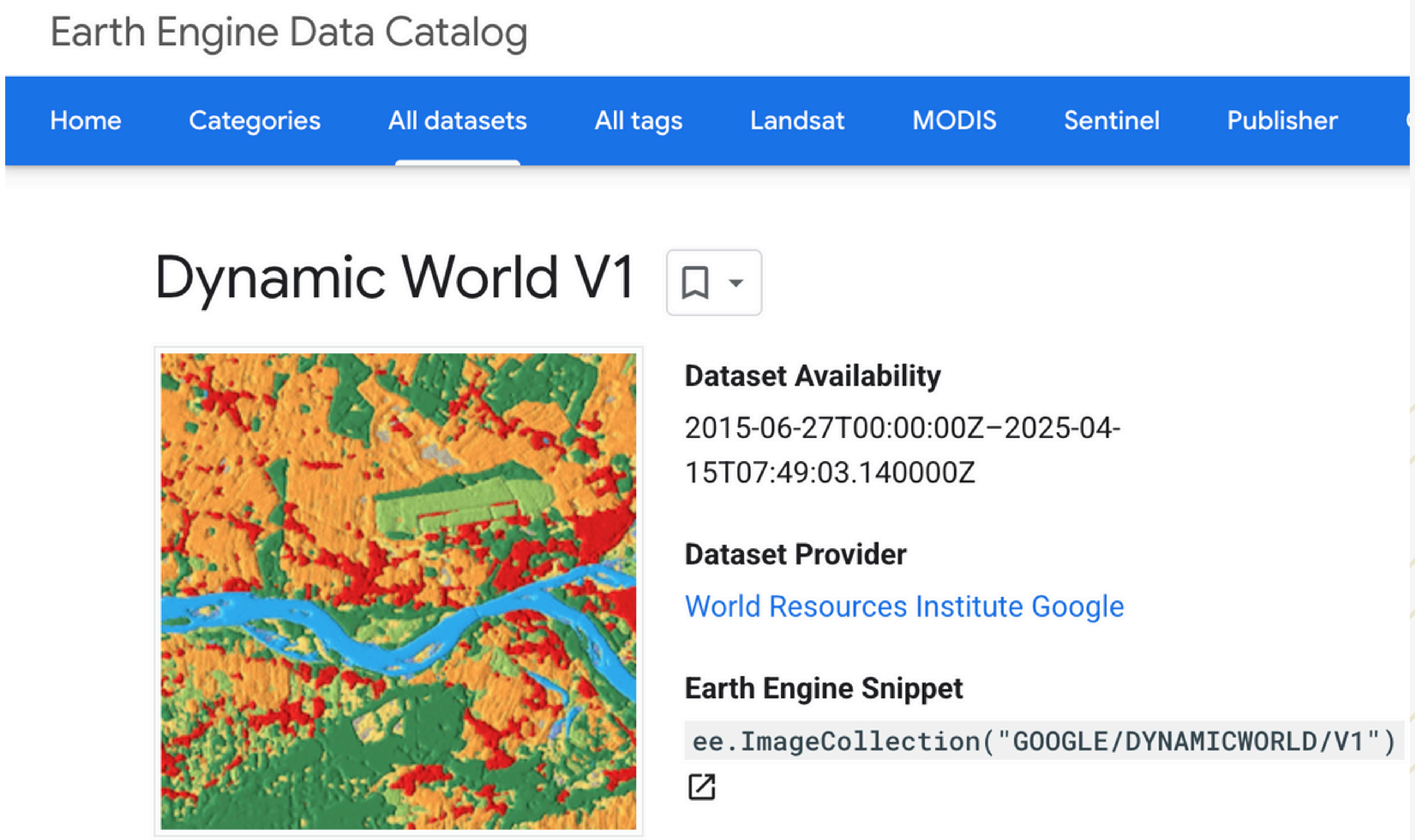
- p_o = observed agreement (same as overall accuracy A_o),
- p_c = expected chance agreement
- *pi+* is the proportion of observations in the predicted class *i*, and *p+i* is the proportion in the actual (reference) class *i*

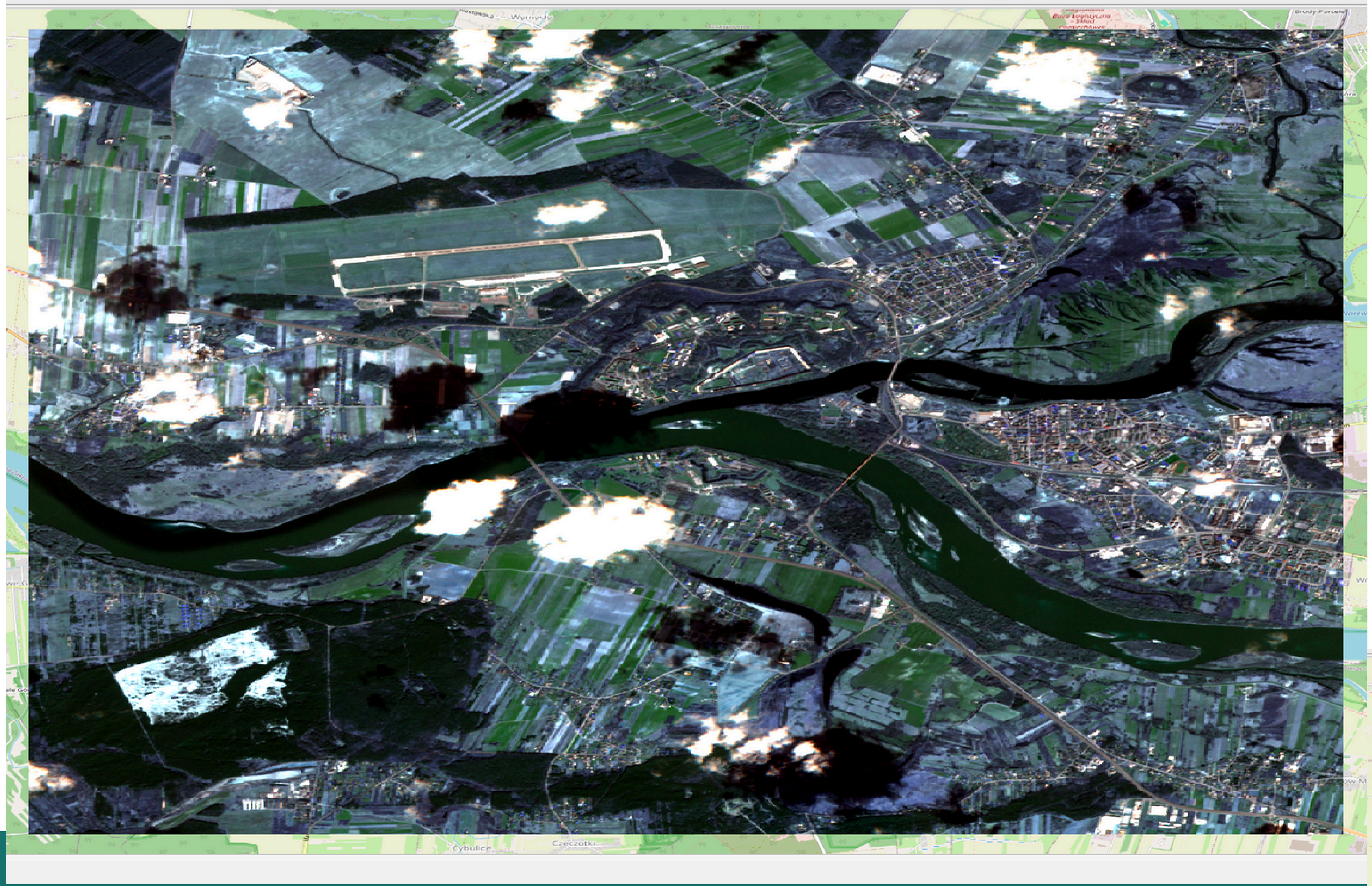
DATASET OVERVIEW

The study employs the Dynamic World Version 1 (DW V1) dataset, a land use/land cover (LULC) classification product. Jointly developed by Google and the World Resources Institute (WRI), this dataset leverages deep learning to generate near real-time predictions from satellite imagery.

Key features of DW V1:

- Based on Sentinel-2 Level-1C imagery
- Provides 10-meter spatial resolution LULC maps
- Delivers near real-time predictions
- Utilizes a deep learning model trained on over 24,000 globally annotated points
- Offers global coverage with high temporal frequency





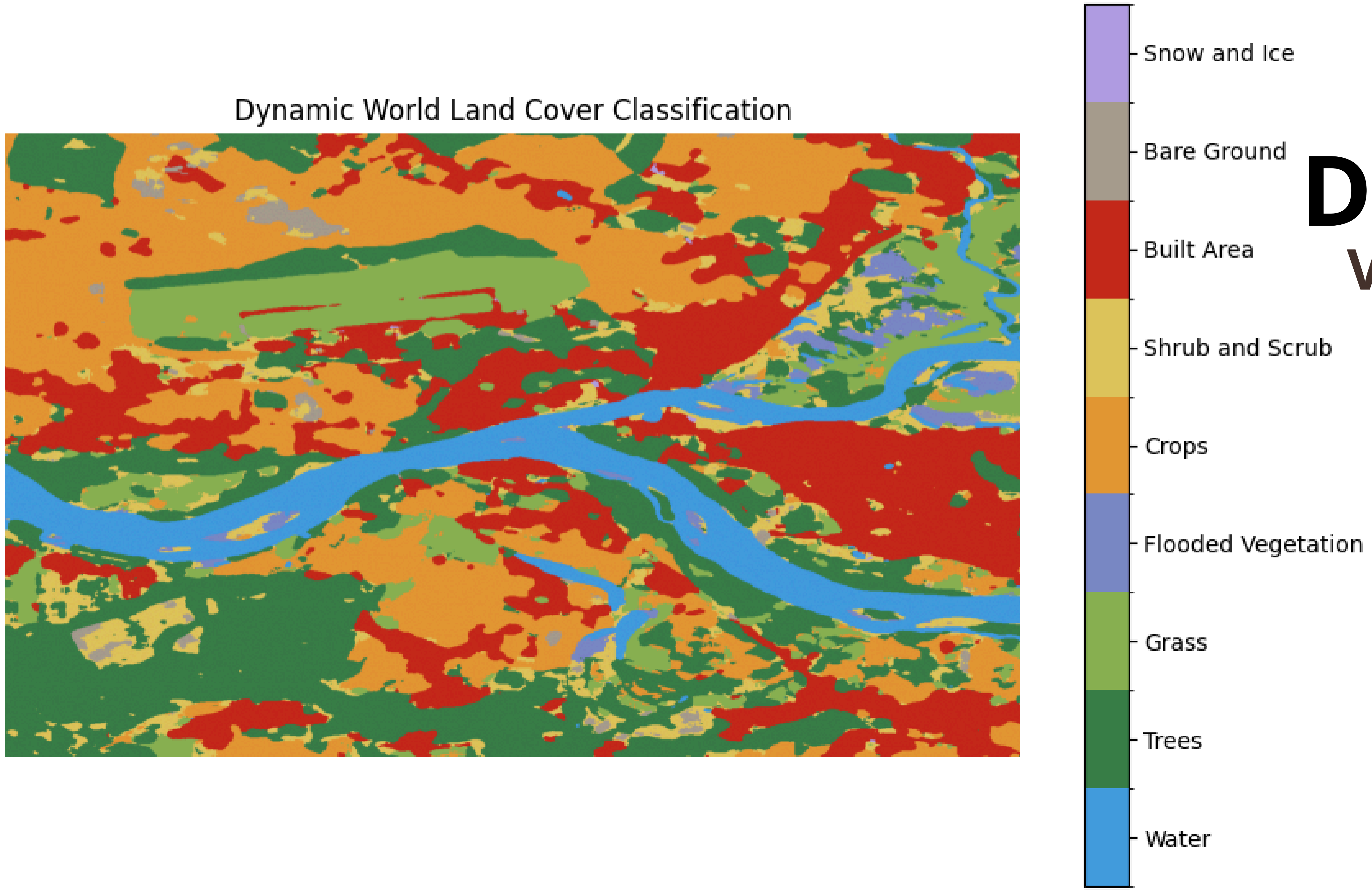
DATASET OVERVIEW

VISUALIZATION OF THE DATASET

The satellite image and the corresponding class label image from the Rater dataset were uploaded and visualized in QGIS. The following snippets illustrate the spatial distribution and land cover classification as rendered in the software:

DATASET OVERVIEW

VISUALIZATION OF THE DATASET

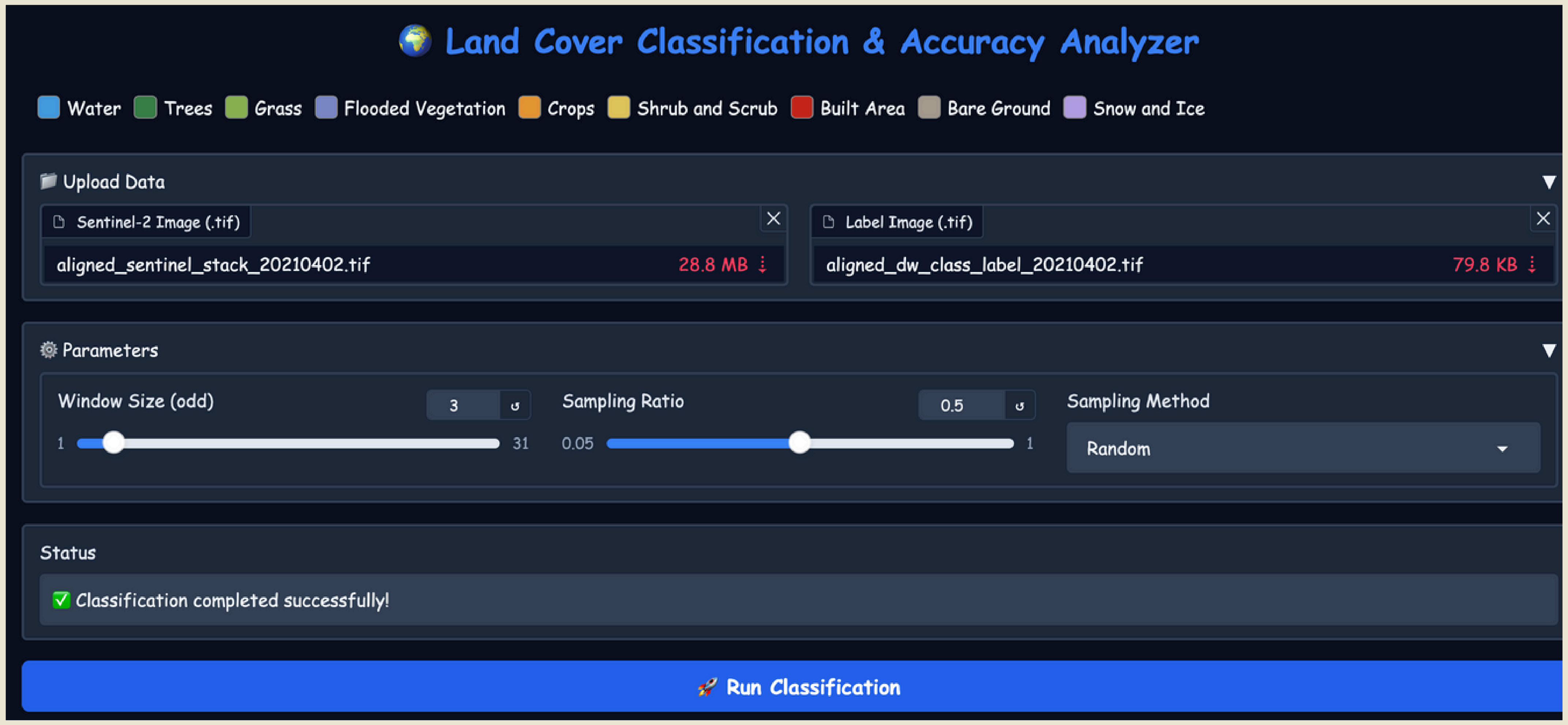


The above is the image of the raster data when the Label class table was followed.

Label Class table as in the Website

Value	Color	Description
0	#419bdf	water
1	#397d49	trees
2	#88b053	grass
3	#7a87c6	flooded_vegetation
4	#e49635	crops
5	#dfc35a	shrub_and_scrub
6	#c4281b	built
7	#a59b8f	bare
8	#b39fe1	snow_and_ice

METHODOLOGY
GRADIO INTERFACE & TOOLS



File Uploads

Two raster images are uploaded through the GUI:

- Sentinel-2 Image (.tif):

A multiband GeoTIFF containing spectral bands from Sentinel-2 satellite, used as the primary input for classification.

- Label Image (.tif):

A single-band raster with class labels (e.g., Water, Trees, Grass, Urban) serving as ground truth.

Parameter Configuration

Users can interactively set the following parameters:

- **Window Size (int):**

Defines spatial context window (e.g., 3, 5, 7); larger sizes capture more neighborhood information.

- **Sampling Ratio (float, 0–1):**

Specifies the fraction of valid labeled pixels to use (e.g., 0.6 → 60% of labeled data used).

- **Sampling Method:**
  - Random: Uniform sampling across valid pixels
  - Stratified: Proportional sampling based on class distribution (helps with class balance)

THE BIG PICTURE

WHAT DOES THE CLASSIFY() FUNCTION DO?

- Preprocessing
- Sample extraction
- Training/testing split
- Model training
- Evaluation using accuracy metrics and confusion matrix
- Reporting the results in both tabular and visual formats

METHODOLOGY

CLASSIFY() FUNCTION

We would now go in depth of the Classify function in our analysis:

LOADING OF THE IMAGE

```
def classify(s2_img_file, label_img_file, window_size, ratio, method):
    s2_img, s2_profile, _ = read_raster(s2_img_file.name)
    labels_img, label_profile, class_data = read_raster(label_img_file.name)
    height, width = class_data.shape
    pixel_size = abs(label_profile['transform'][0])
```

- Loads the Sentinel-2 image and label raster into memory.
- read__raster returns the image array, its metadata profile, and the pixel data (class__data).

SAMPLE PIXELS

```
X_sampled, y_sampled = sample_pixels(X, y, ratio, method)
```

- Reduces the number of samples according to the user-specified ratio and method.
- Could be random or stratified

FEATURE REPRESENTATION: WINDOW-BASED FEATURES

- A square window (e.g., 3x3, 5x5) is applied around each pixel
- All values in the window across all spectral bands are flattened into a single feature vector
- Captures spatial context, enhancing classification in heterogeneous areas
- X: feature vectors (flattened window of image bands).
- y: corresponding class labels for the center pixel in each window.

```
X, y = extract_window_samples(s2_img, class_data, window_size)
if X.shape[0] == 0:
    return "No samples extracted, please check inputs."
```

METHODOLOGY

SAMPLE EDA

Purpose of EDA

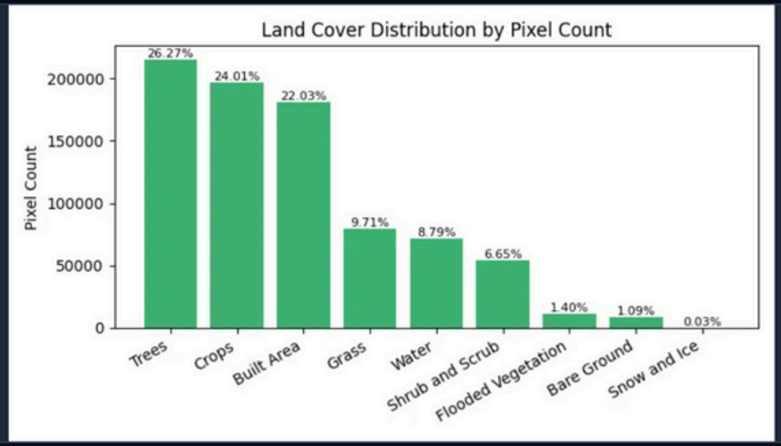
- Understanding the distribution of classes in the sampled dataset
- Analyzing the spectral and spatial patterns of each land cover class
- Identifying issues like class imbalance, outliers, or noisy labels

Sampled EDA | Classification Results

Sampled EDA Table

Class ID	Class Name	Pixel Count	Area (m²)	Percentage (%)
1	Trees	215310	0.0017374878599019001	26.27490844480024
4	Crops	196771	0.0015878836267742176	24.012540103069007
6	Built Area	180534	0.0014568558510962316	22.031091547877786
2	Grass	79570	0.0006421063072425535	9.710159606858738
0	Water	72023	0.0005812042549519973	8.789177144209965
5	Shrub and Scrub	54519	0.00043995216494353113	6.6531128767919006
3	Flooded Vegetation	11494	0.00009275317199253374	1.4026464059473964
7	Bare Ground	8951	0.00007223191600010174	1.092316685195332
8	Snow and Ice	279	0.0000022514472756148535	0.034047185249636645

Sampled EDA Plot



Download EDA Plot

eda_plot.png 31.0 KB

Use via API · Built with Gradio · Settings

```python
eda_df_sampled, eda_fig_sampled = generate_eda(y_sampled.reshape(-1, 1), pixel_size)
```


Class ID	Class Name	Pixel Count	Area (m²)	Percentage (%)
1	Trees	215310	0.0017374878599019001	26.27490844480024
4	Crops	196771	0.0015878836267742176	24.012540103069007
6	Built Area	180534	0.0014568558510962316	22.031091547877786
2	Grass	79570	0.0006421063072425535	9.710159606858738
0	Water	72023	0.0005812042549519973	8.789177144209965
5	Shrub and Scrub	54519	0.00043995216494353113	6.6531128767919006
3	Flooded Vegetation	11494	0.00009275317199253374	1.4026464059473964
7	Bare Ground	8951	0.00007223191600010174	1.092316685195332
8	Snow and Ice	279	0.0000022514472756148535	0.034047185249636645

CLASS DISTRIBUTION HISTOGRAM

- Displays a bar plot showing the number of sampled pixels per class
- Assists in evaluating whether sampling was balanced, especially for stratified sampling
- Highlights any skewed distribution that might impact classification performance

METHODOLOGY

CLASSIFY()

The classification module is the core component of the GUI. It uses supervised learning to assign land cover classes to each pixel, leveraging both spectral and spatial context.

Model Used

- Non-parametric & robust to noise
- Effective for high-dimensional data
- Suitable for small to medium datasets
- Provides feature importance insights

Implementation Details

- Library: scikit-learn
- Parameters:
- n_estimators=100
- random_state=42

TRAINING ON SAMPLED PIXELS

- Uses training mask to extract sampled pixels
- Features and labels are derived from sampled windows
- Split into 80-20 train-test set (stratified) for training

```
clf = RandomForestClassifier(n_estimators=100, max_depth=30, n_jobs=-1)
```

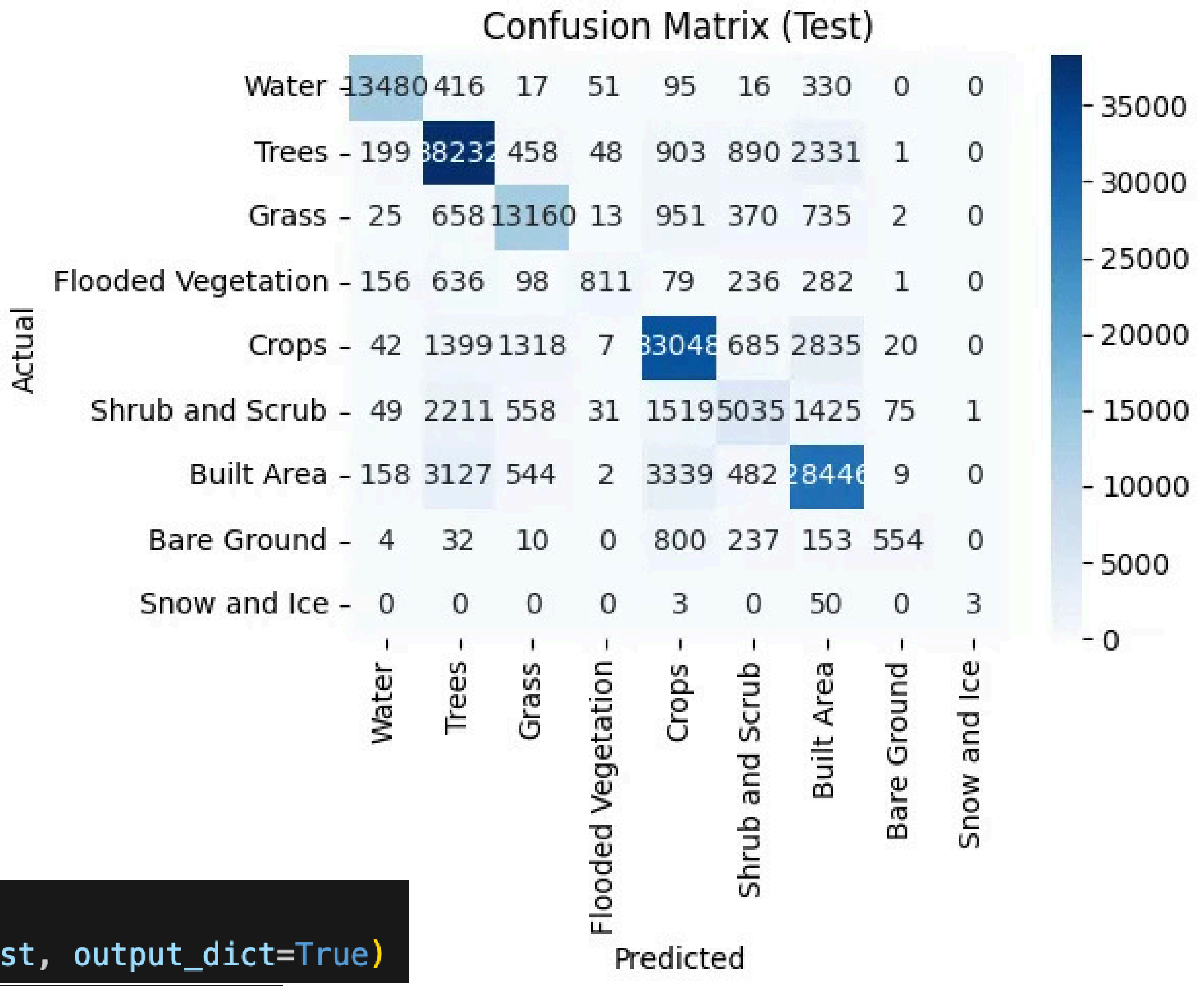
- **n_estimators=100:** Builds 100 decision trees in the ensemble.
- **max_depth=30:** Limits the maximum depth of each tree to 30, controlling model complexity and reducing overfitting.
- **n_jobs=-1:** Utilizes all available CPU cores to parallelize the training process for improved performance.

```
X_sampled, y_sampled = sample_pixels(X, y, ratio, method)
eda_df_sampled, eda_fig_sampled = generate_eda(y_sampled.reshape(-1, 1), pixel_size)
X_train, X_test, y_train, y_test = train_test_split(X_sampled, y_sampled, stratify=y_sampled, test_size=0.2, random_state=42)
```

METHODOLOGY
CLASSIFY()

Classification of Test Set Pixels

- After training, the Random Forest classifier is applied to the test set of sampled pixels.
- Uses the same window-based feature extraction as in training.
- Ensures consistency between training and testing data representations.
- Results are used for accuracy assessment and confusion matrix generation.




```python
y_pred_test = clf.predict(X_test)
test_report = classification_report(y_test, y_pred_test, output_dict=True)
```

```python
cm = confusion_matrix(y_test, y_pred_test)
cm_fig, ax = plt.subplots(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', ax=ax,
            xticklabels=list(class_map.values()), yticklabels=list(class_map.values()))
ax.set_title("Confusion Matrix (Test)")
ax.set_xlabel("Predicted")
ax.set_ylabel("Actual")
plt.tight_layout()
```

METHODOLOGY
CLASSIFICATION REPORT

Classification Report

Type	precision	recall	f1-score	support
Class 0	0.955	0.936	0.945	14405
Class 1	0.818	0.888	0.852	43062
Class 2	0.814	0.827	0.821	15914
Class 3	0.842	0.353	0.497	2299
Class 4	0.811	0.84	0.825	39354
Class 5	0.633	0.462	0.534	10904
Class 6	0.777	0.788	0.783	36107
Class 7	0.837	0.309	0.452	1790
Class 8	0.75	0.054	0.1	56
Accuracy	0.81	0.81	0.81	0.81
Macro Average	0.804	0.606	0.645	163891
Weighted Average	0.807	0.81	0.805	163891

METHODOLOGY
ACCURACY ASSESMENT

Evaluates classifier performance by comparing predicted labels with the reference ground truth map.

- **Overall Accuracy**
- **User's Accuracy (Precision)**
- **Producer's Accuracy (Recall)**
- **Kappa Coefficient**
- **Tau Statistic**
- **Z-Statistic & Sigma Kappa**

```python
def compute_detailed_accuracy(cm):
    n_classes = cm.shape[0]
    total = np.sum(cm)
    pi_plus = cm.sum(axis=1) / total
    pj_plus = cm.sum(axis=0) / total
    user_acc = np.diag(cm) / cm.sum(axis=1)
    prod_acc = np.diag(cm) / cm.sum(axis=0)
    si_user = np.sqrt(user_acc * (1 - user_acc) / cm.sum(axis=1))
    si_prod = np.sqrt(prod_acc * (1 - prod_acc) / cm.sum(axis=0))
    ci_user = [(round(u - 1.96*s, 4), round(u + 1.96*s, 4)) for u, s in zip(user_acc, si_user)]
    ci_prod = [(round(u - 1.96*s, 4), round(u + 1.96*s, 4)) for u, s in zip(prod_acc, si_prod)]
    oa = np.trace(cm) / total
    po = oa
    pe = np.sum(np.sum(cm, axis=0) * np.sum(cm, axis=1)) / (total ** 2)
    kappa = (po - pe) / (1 - pe) if pe != 1 else None
    sigma_kappa = np.sqrt((po * (1 - po)) / (total * (1 - pe)**2)) if pe != 1 else None
    z = (kappa / sigma_kappa) if sigma_kappa else None
    tau = (po - pe) / (1 - pe) if pe != 1 else None
    overall_df = pd.DataFrame({
        'Metric': ['Overall Accuracy', 'Kappa', 'Sigma Kappa', 'Tau', 'Z-Statistic'],
        'Value': [round(oa, 4), round(kappa, 4), round(sigma_kappa, 4) if sigma_kappa else '',
                  round(tau, 4) if tau else '', round(z, 4) if z else '']
    })
    return overall_df
```

METHODOLOGY

HTML OUTPUT: EXPANDED CONFUSION MATRIX

	Water	Trees	Grass	Flooded Vegetation	Crops	Shrub and Scrub	Built Area	Bare Ground	Snow and Ice	Total	pi+	Ci	si	95% CI of Ci
Water	13480.000000	416.000000	17.000000	51.000000	95.000000	16.000000	330.000000	0.000000	0.000000	14405	0.087900	0.935800	0.002000	0.9318 - 0.9398
Trees	199.000000	38232.000000	458.000000	48.000000	903.000000	890.000000	2331.000000	1.000000	0.000000	43062	0.262700	0.887800	0.001500	0.8849 - 0.8908
Grass	25.000000	658.000000	13160.000000	13.000000	951.000000	370.000000	735.000000	2.000000	0.000000	15914	0.097100	0.826900	0.003000	0.8211 - 0.8328
Flooded Vegetation	156.000000	636.000000	98.000000	811.000000	79.000000	236.000000	282.000000	1.000000	0.000000	2299	0.014000	0.352800	0.010000	0.3332 - 0.3723
Crops	42.000000	1399.000000	1318.000000	7.000000	33048.000000	685.000000	2835.000000	20.000000	0.000000	39354	0.240100	0.839800	0.001800	0.8361 - 0.8434
Shrub and Scrub	49.000000	2211.000000	558.000000	31.000000	1519.000000	5035.000000	1425.000000	75.000000	1.000000	10904	0.066500	0.461800	0.004800	0.4524 - 0.4711
Built Area	158.000000	3127.000000	544.000000	2.000000	3339.000000	482.000000	28446.000000	9.000000	0.000000	36107	0.220300	0.787800	0.002200	0.7836 - 0.792
Bare Ground	4.000000	32.000000	10.000000	0.000000	800.000000	237.000000	153.000000	554.000000	0.000000	1790	0.010900	0.309500	0.010900	0.2881 - 0.3309
Snow and Ice	0.000000	0.000000	0.000000	0.000000	3.000000	0.000000	50.000000	0.000000	3.000000	56	0.000300	0.053600	0.030100	-0.0054 - 0.1125
Total	14113.000000	46711.000000	16163.000000	963.000000	40737.000000	7951.000000	36587.000000	662.000000	4.000000	163891	1.000000			
Producer's reliability (Oj)	0.955100	0.818500	0.814200	0.842200	0.811300	0.633300	0.777500	0.836900	0.750000					
si (Prod)	0.001700	0.001800	0.003100	0.011700	0.001900	0.005400	0.002200	0.014400	0.216500					
95% CI of Oj	0.9517 - 0.9586	0.815 - 0.822	0.8082 - 0.8202	0.8191 - 0.8652	0.8075 - 0.8151	0.6227 - 0.6438	0.7732 - 0.7818	0.8087 - 0.865	0.3256 - 1.1744					

METHODOLOGY
EXCEL AND HTML OUTPUTS

```python
overall_acc_df = compute_detailed_accuracy(cm)
excel_path, styled_matrix_html = save_confusion_matrix_excel(
    cm=cm,
    oa=overall_acc_df.loc[0, 'Value'],
    kappa=overall_acc_df.loc[1, 'Value'],
    sigma_kappa=overall_acc_df.loc[2, 'Value'],
    tau=overall_acc_df.loc[3, 'Value'],
    z=overall_acc_df.loc[4, 'Value'],
    class_names=list(class_map.values())
)
```

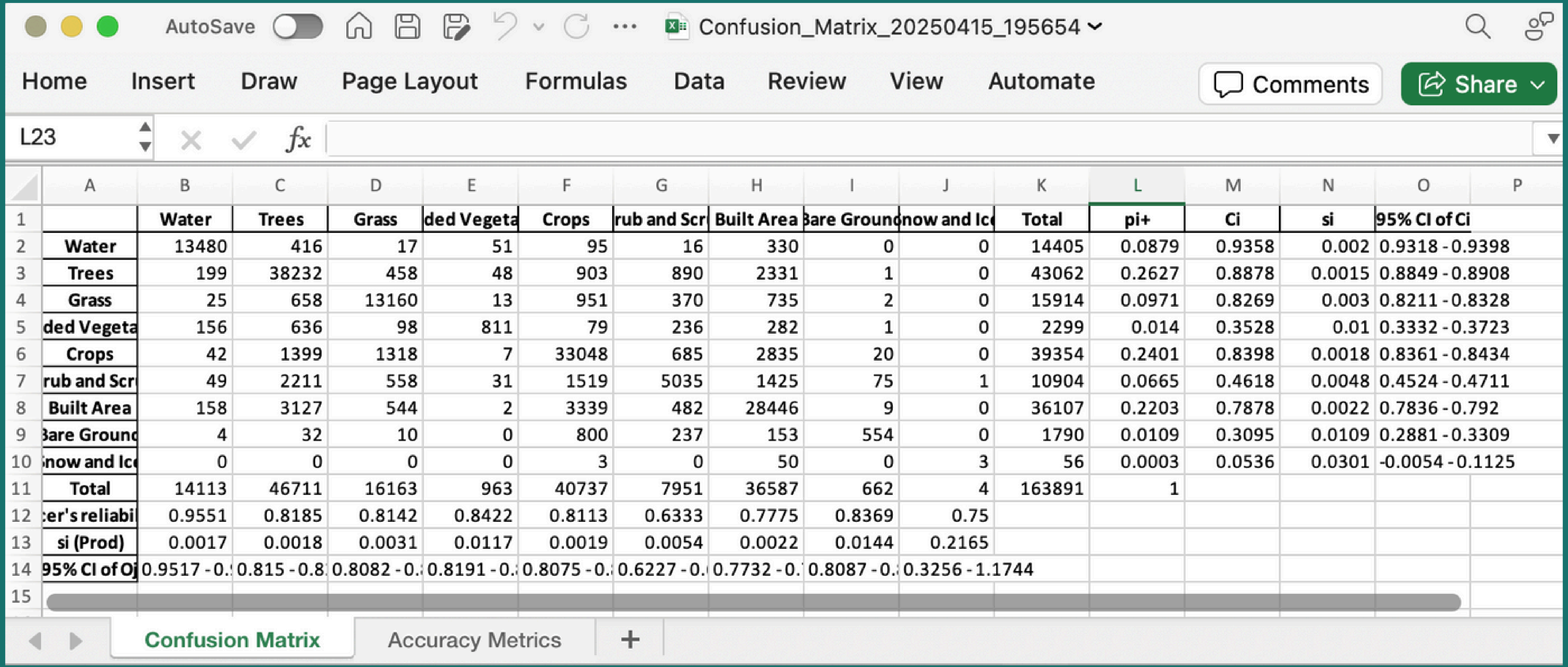
Overall Accuracy Metrics

Metric	Value
Overall Accuracy	0.8101
Kappa	0.7614
Sigma Kappa	0.0012
Tau	0.7614
Z-Statistic	625.3417

Download Accuracy Report (.xlsx)

Confusion_Matrix_20250415_195654.xlsx 6.8 KB

The overall accuracy and other key metrics are clearly displayed on the interface, and the results can be easily exported in Excel format.



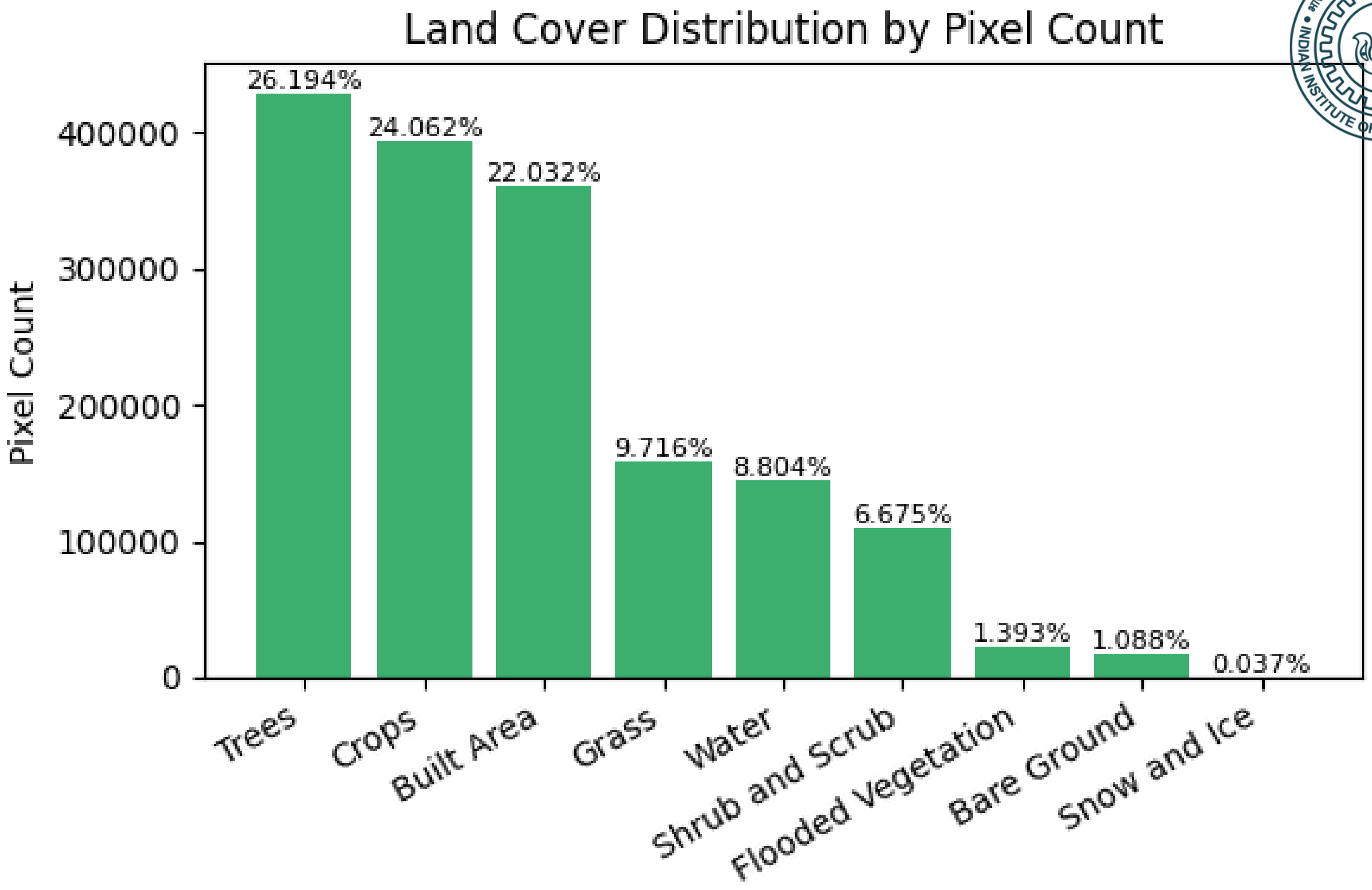
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1		Water	Trees	Grass	ded Vegeta	Crops	rub and Scr	Built Area	Bare Ground	now and Ice	Total	pi+	Ci	si	95% CI of Ci
2	Water	13480	416	17	51	95	16	330	0	0	14405	0.0879	0.9358	0.002	0.9318 - 0.9398
3	Trees	199	38232	458	48	903	890	2331	1	0	43062	0.2627	0.8878	0.0015	0.8849 - 0.8908
4	Grass	25	658	13160	13	951	370	735	2	0	15914	0.0971	0.8269	0.003	0.8211 - 0.8328
5	ded Vegeta	156	636	98	811	79	236	282	1	0	2299	0.014	0.3528	0.01	0.3332 - 0.3723
6	Crops	42	1399	1318	7	33048	685	2835	20	0	39354	0.2401	0.8398	0.0018	0.8361 - 0.8434
7	rub and Scr	49	2211	558	31	1519	5035	1425	75	1	10904	0.0665	0.4618	0.0048	0.4524 - 0.4711
8	Built Area	158	3127	544	2	3339	482	28446	9	0	36107	0.2203	0.7878	0.0022	0.7836 - 0.792
9	Bare Ground	4	32	10	0	800	237	153	554	0	1790	0.0109	0.3095	0.0109	0.2881 - 0.3309
10	now and Ice	0	0	0	0	3	0	50	0	3	56	0.0003	0.0536	0.0301	-0.0054 - 0.1125
11	Total	14113	46711	16163	963	40737	7951	36587	662	4	163891	1			
12	cer's reliabi	0.9551	0.8185	0.8142	0.8422	0.8113	0.6333	0.7775	0.8369	0.75					
13	si (Prod)	0.0017	0.0018	0.0031	0.0117	0.0019	0.0054	0.0022	0.0144	0.2165					
14	95% CI of O	0.9517 - 0.	0.815 - 0.8	0.8082 - 0.	0.8191 - 0.	0.8075 - 0.	0.6227 - 0.	0.7732 - 0.	0.8087 - 0.	0.3256 - 1.1744					
15															

RESULTS

AND DISCUSSIONS

EXPERIMENTAL SETUP

The land cover classification was performed using Sentinel-2 imagery and labeled data for the study area. The classification algorithm employed a Random Forest classifier, and performance was evaluated across varying parameters: **window sizes, sampling methods, and sampling ratios.**

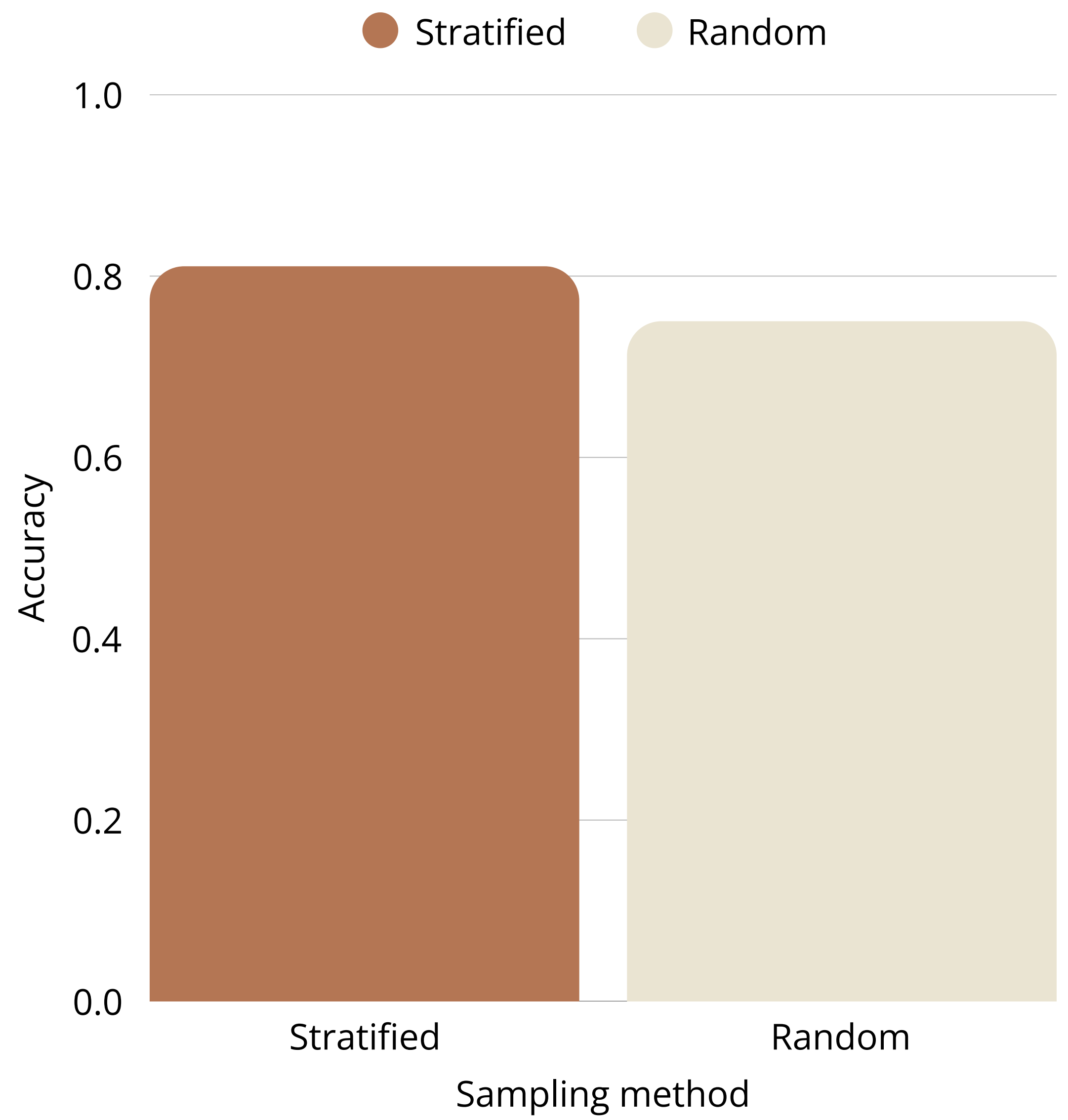


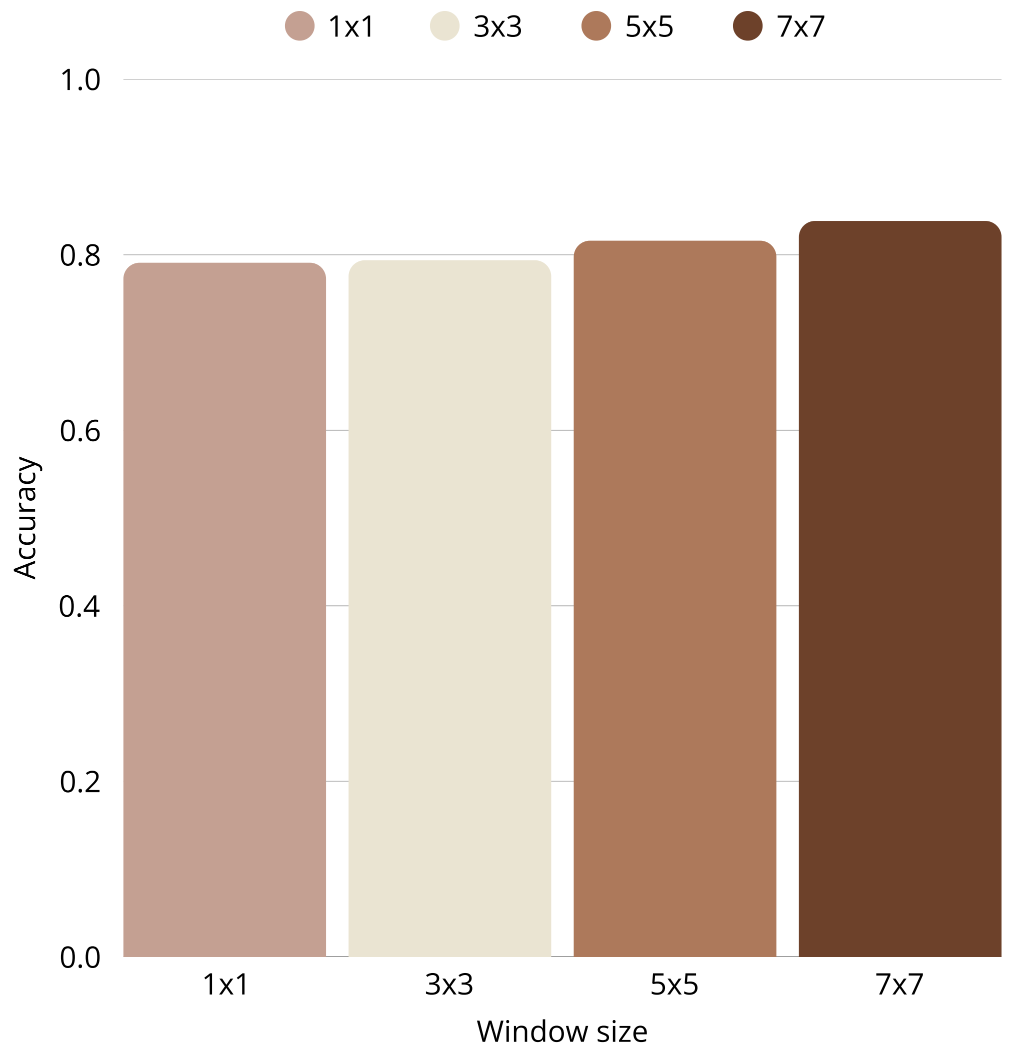
Land Cover EDA Summary

	Class ID	Class Name	Pixel Count	Area (m^2)	Percentage (%)
1	1	Trees	429292	0.003464	26.193879
0	4	Crops	394356	0.003182	24.062208
4	6	Built Area	361075	0.002914	22.031519
6	2	Grass	159235	0.001285	9.715956
5	0	Water	144286	0.001164	8.803821
3	5	Shrub and Scrub	109395	0.000883	6.674896
8	3	Flooded Vegetation	22826	0.000184	1.392762
2	7	Bare Ground	17835	0.000144	1.088229
7	8	Snow and Ice	602	0.000005	0.036732

Total valid pixels: 1,638,902
Total area: 0.013 m^2

RESULTS AND DISCUSSIONS





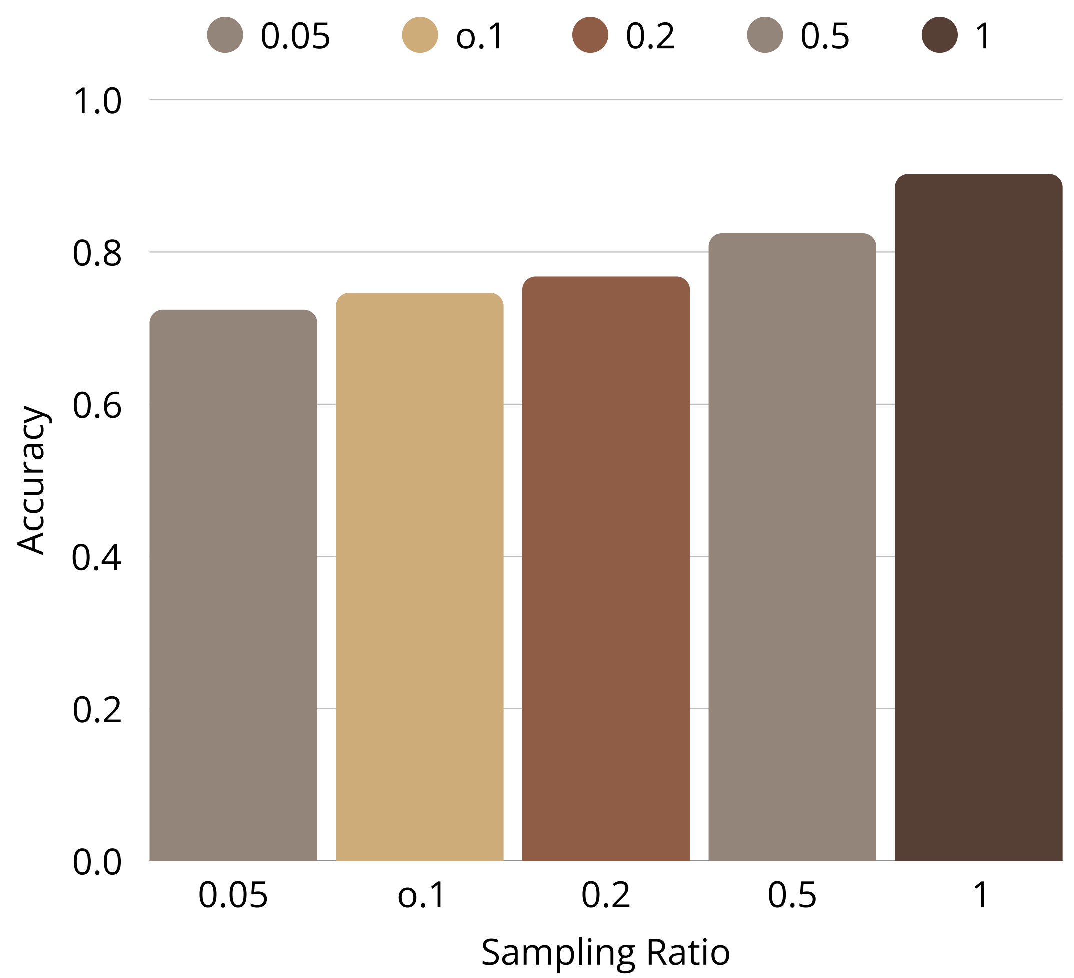
Effect of Sampling Method

- **Stratified sampling** consistently yielded higher classification accuracy.This is because it ensures balanced representation across all land cover classes.
- **Random sampling**, while faster, resulted in class imbalance, leading to reduced performance for underrepresented classes like 'Snow and Ice' or 'Flooded Vegetation'.

Influence of Window Size

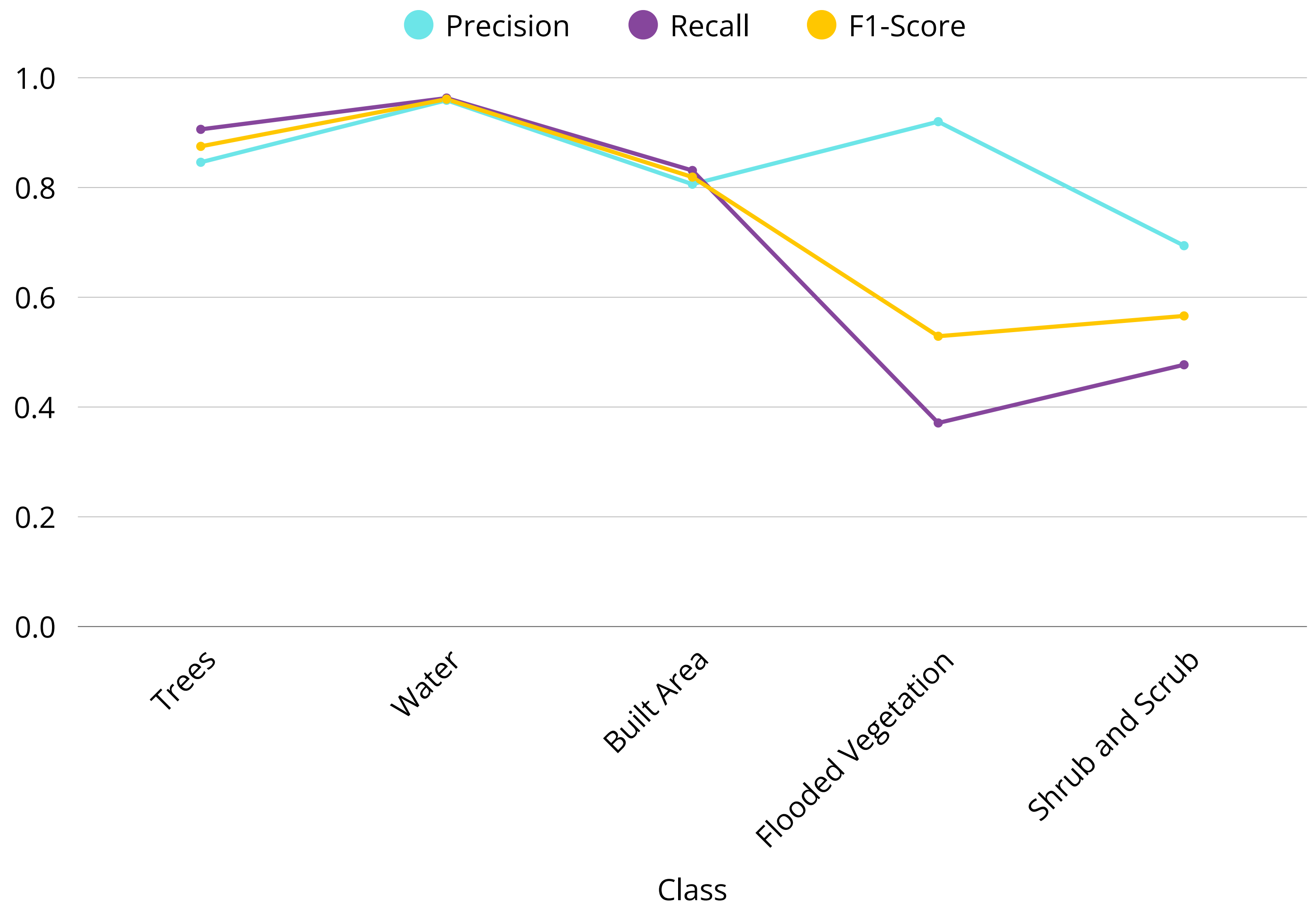
- A moderate window size (5 or 7) provided the best results, balancing context and local features.
- Larger windows (e.g., 21) began to introduce noise and redundant information, slightly reducing performance.
- Small windows (3) often lacked spatial patterns, leading to reduced classification accuracy.

RESULTS AND DISCUSSIONS



Sampling Ratio Impact

- Sampling ratio controls the proportion of labeled pixels used for training. Higher ratios provide more training data.
- The results show that increasing the sampling ratio from 0.1 to 0.5 leads to a modest improvement in overall accuracy (0.7464-0.8246). This trend is expected, as more training samples generally allow the classifier to learn more robust decision boundaries



Class-wise Performance

From the classification report and confusion matrix:

- High accuracy was achieved for dominant classes such as Trees, Water, and Built Areas.
- Misclassifications commonly occurred between Grass and Crops and between Bare Ground andBuilt Area

INTERPRETATION AND RECOMMENDATIONS

0.7242 - 0.9025	Overall accuracy
0.7096 - 0.7969	Kappa coefficient

- Larger window sizes and higher sampling ratios generally improve classification accuracy and statistical confidence, but at the cost of increased computation.
- Stratified sampling is recommended for smaller training sets to ensure all classes are represented, while random sampling can be effective when using a larger proportion of the data.
- The classifier performs robustly across a range of parameter settings, with only modest variations in accuracy and kappa, indicating the method's stability for land cover mapping tasks.

LIMITATIONS

Dependence on Reference Labels
The code requires a reference label image for supervised classification and accuracy assessment. If the reference data is inaccurate, outdated, or misaligned with the Sentinel-2 image, the classification results and accuracy metrics will be unreliable.

Limited to Sentinel-2 and .tif Format
The current implementation is tailored for Sentinel-2 imagery in .tif format. It does not natively support other satellite data sources (e.g., Landsat, MODIS) or file formats, limiting its applicability.

Fixed Class Definitions
The class labels and color mappings are hardcoded for nine specific land cover types. The system cannot easily adapt to different classification schemes or a different number of classes without code modification.

Random Forest Only
The code uses a Random Forest classifier with fixed hyperparameters. There is no option for users to select other algorithms (e.g., SVM, neural networks) or to tune hyperparameters, which may limit classification performance in some scenarios.

Computational Efficiency
For large images or large window sizes, the extraction of windowed samples and training can be computationally intensive and memory-demanding, potentially limiting scalability to very large datasets.

No Handling of Missing Data or Clouds
There is no explicit handling of missing data, cloud cover, or other common remote sensing artifacts, which can affect classification accuracy.

SCOPE FOR FUTURE WORK

Support for Multiple Data Sources and Formats
Extend the code to handle other satellite sensors (e.g., Landsat, MODIS) and additional file formats, increasing its versatility for different remote sensing applications.

Dynamic Class Definitions
This allows users to define their own class labels and color schemes through the interface, making the tool adaptable to various land cover classification schemes.

Algorithm Selection and Hyperparameter Tuning:
Provide options for users to select different classification algorithms (e.g., SVM, k-NN, neural networks) and tune their hyperparameters via the interface.

Scalability and Performance Optimization
Optimize the code for large-scale processing, possibly by:
Implementing batch processing or tiling strategies
Leveraging parallel computing or GPU acceleration

Interactive Visualization:
Integrate interactive map viewers (e.g., using folium or ipyleaflet) to allow users to explore classification results spatially and compare with original imagery.

Automated Accuracy Assessment:
Implement more advanced accuracy assessment techniques, such as cross-validation, bootstrapping, or spatially stratified sampling, to provide more reliable performance metrics.

CONCLUSION

- This study applies **Random Forest (RF)** for land cover classification using **Sentinel-2 imagery**, emphasizing the impact of **window size, sampling ratio**, and **method** on classification accuracy.
- Results show that a **moderate spatial window** and a **balanced training set** significantly improve performance for major classes like ‘**Trees**’ and ‘**Water**’
- Challenges observed: **Class imbalance, Spectral similarity** among certain classes, **Limited generalizability**
- Highlights the need for:Integration of **multi-temporal** and **multi-sensor data**, **Advanced ML models** and feature engineering
- The approach offers a **flexible, scalable, and open-source solution** for environmental monitoring and urban planning.
- With further development, this method supports **accurate and timely land cover mapping** for sustainable decision-making.

REFERENCES

[1] R. G. Congalton, “A review of assessing the accuracy of classifications of remotely sensed data,” Remote Sensing of Environment, vol. 37, no. 1, pp. 35–46, 1991. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/003442579190048B

[2] M. Story and R. G. Congalton, “Accuracy assessment: A user’s perspective,” Photogrammetric Engineering and Remote Sensing, vol. 52, no. 3, pp. 397–399, 1986. [Online]. Available: https://www.asprs.org/wp-content/uploads/pers/1986journal/mar/1986_mar_397-399.pdf

[3] G. M. Foody, “Status of land cover classification accuracy assessment,” Remote Sensing of Environment, vol. 80, no. 1, pp. 185–201, 2002. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0034425701002954

[4] R. G. Congalton and K. Green, Assessing the Accuracy of Remotely Sensed Data: Principles and Practices, CRC Press, 1999. [Online]. Available: https://www.taylorfrancis.com/books/mono/10.1201/9780429052729

[5] Congalton, R. G. (1991). A review of assessing the accuracy of classifications of remotely sensed data. Remote Sensing of Environment, 37(1), 35–46. https://doi.org/10.1016/0034-4257(91)90048-B

[6] Foody, G. M. (2002). Status of land cover classification accuracy assessment. Remote Sensing of Environment, 80(1), 185–201. https://doi.org/10.1016/S0034-4257(01)00295-4

[7] Olofsson, P., et al. (2014). Good practices for estimating area and assessing accuracy of land change. Remote Sensing of Environment, 148, 42–57. https://doi.org/10.1016/j.rse.2014.02.015

[8] Zhang, C., et al. (2020). An object-based approach to land cover classification using Sentinel-2 data. Remote Sensing, 12(4), 626. https://doi.org/10.3390/rs12040626

REFERENCES

[9] Buchhorn, M., et al. (2020). Copernicus Global Land Cover Layers—Collection 2. Remote Sensing, 12(6), 1044. https://doi.org/10.3390/rs12061044

[10] Google. (2021). Dynamic World - Near real-time land use classification. https://dynamicworld.app

[11] Congalton, R. G., & Mead, R. A. (1983). A quantitative method to test for consistency and correctness in photointerpretation. Photogrammetric Engineering and Remote Sensing, 49(1), 69–74. https://www.asprs.org/wp-content/uploads/pers/1983journal/jan/1983_jan_69-74.pdf:contentReference[oaicite:1]{index=1}

[12] Congalton, R. G., & Green, K. (2009). Assessing the Accuracy of Remotely Sensed Data: Principles and Practices(2nd ed.). CRC Press. https://www.taylorfrancis.com/books/mono/10.1201/9780429052729/assessing-accuracy-remotely-sensed-data-russell-congalton-kass-green:contentReference[oaicite:3]{index=3}

[13] Cohen, J. (1960). A coefficient of agreement for nominal scales. Educational and Psychological Measurement, 20(1), 37–46. https://doi.org/10.1177/001316446002000104:contentReference[oaicite:5]{index=5}

[14] Cohen, J. (1960). A coefficient of agreement for nominal scales. Educational and Psychological Measurement, 20(1), 37–46. https://doi.org/10.1177/001316446002000104

[15] Rosenfield, G. H., & Fitzpatrick-Lins, K. (1986). A coefficient of agreement as a measure of thematic classification accuracy. Photogrammetric Engineering and Remote Sensing, 52(2), 223–227. https://www.asprs.org/wp-content/uploads/pers/1986journal/feb/1986_feb_223-227.pdf

[16] Ma, Z., & Redmond, R. L. (1995). Tau coefficients for accuracy assessment of classification of remote sensing data. Photogrammetric Engineering and Remote Sensing, 61(4), 435–439. https://www.asprs.org/wp-content/uploads/pers/1995journal/apr/1995_apr_435-439.pdf

REFERENCES

[17] Næsset, E. (1996). Use of the τ coefficient in accuracy assessment of maps derived from remotely sensed data. International Journal of Remote Sensing, 17(6), 1183–1190. https://doi.org/10.1080/01431169608949067

[18] Rossiter, D. G. (2001). A theoretical framework for land evaluation. Geoderma, 90(3-4), 207–228. https://doi.org/10.1016/S0016-7061(99)00048-0

[19] Zhu, X. X., & Tuia, D. et al. (2017). Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. IEEE Geoscience and Remote Sensing Magazine, 5(4), 8–36. https://doi.org/10.1109/MGRS.2017.2762307

[20] Congalton, R. G. (1991). A review of assessing the accuracy of classifications of remotely sensed data. Remote Sensing of Environment, 37(1), 35–46.https://doi.org/10.1016/0034-4257(91)90048-B

[21] Foody, G. M. (2002). Status of land cover classification accuracy assessment. Remote Sensing of Environment, 80(1), 185–201. https://doi.org/10.1016/S0034-4257(01)00295-4

[22] Ma, Z., & Redmond, R. L. (1995). Tau coefficients for accuracy assessment of classification of remote sensing data. Photogrammetric Engineering and Remote Sensing, 61(4), 435–439. https://www.asprs.org/wp-content/uploads/pers/1995journal/apr/1995_apr_435-439.pdf

[23] Stehman, S. V. (1997). Selecting and interpreting measures of thematic classification accuracy. Remote Sensing of Environment, 62(1), 77–89. https://doi.org/10.1016/S0034-4257(97)00083-7

[24] Google Dynamic World Dataset Catalog. *https://developers.google.com/earth-engine/datasets/catalog/GOOGLE_DYNAMICWORLD_V1*

THANK YOU