東京大学大学院新領域創成科学研究科

社会文化環境学専攻

2020 年度

修　士　論　文

Random Forest for Seagrass and Seaweed Habitat Mapping in Suo Nada, the Seto Inland Sea

瀬戸内海周防灘におけるランダムフォレストを用いた藻場分布の把握

2020 年 7 月 10 日提出

指導教員　佐々木淳　教授

Xiaoyu Qin

シャオユ　チン

Table of Contents

Random Forest for Seagrass and Seaweed Habitat Mapping in Suo Nada, the Seto Inland Sea

1 Introduction

1.1 Background

1.1.1 Importance of seagrass and seaweed

Seagrass and seaweed (Fig.1) ecosystems act as primary producers in coastal waters, providing essential functions by producing and exporting organic carbon, regulating carbon dioxide, and storing them inside the water [1]. This means that they support the shallow water area through primary production, but they also play a role in mitigating climate change. Thus, the role of seagrass and seaweed meadows in the coastal marine environment is often compared to that of a land forest and is named by UN Environmental Programme as Blue Carbon (BC) in comparison with terrestrial green carbon [1]. This concept was introduced as a comparative metaphor to highlight that coastal ecosystems, like seagrass, seaweed, mangroves, and salt marshes, contribute significantly to carbon sequestration.

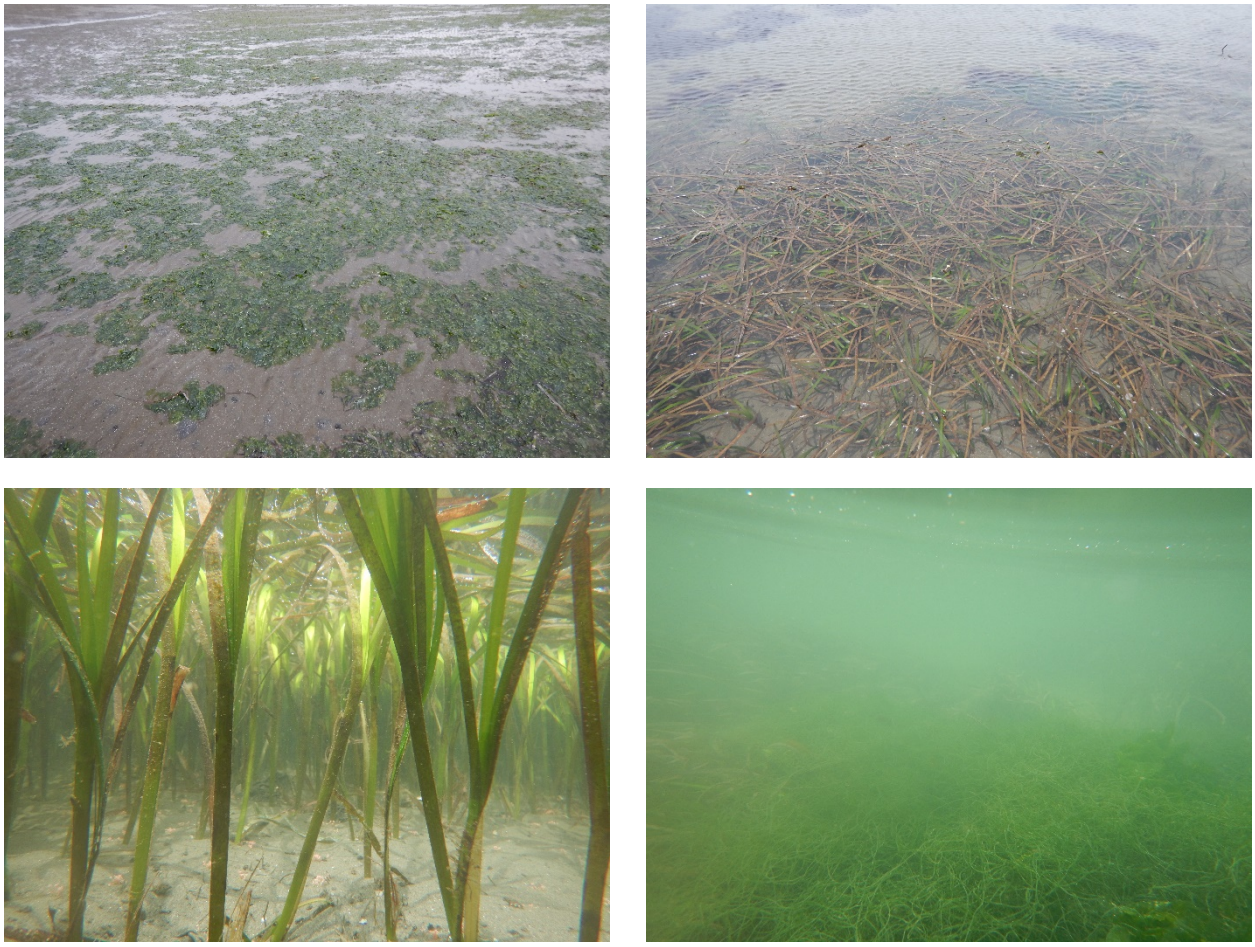


Fig. 1 Zostera Marina beds of Futtsu tidal flat, Tokyo Bay (Source: taken by author)

Although it is important to the environment, seagrass and seaweed are under human pressure. Land reclamation, eutrophication, pollution, and other human activities damage the seabed ecosystems and reduce the benthic biodiversity [2]. For example, in the Seto Inland Sea coastal areas, many tidal flats and seagrass beds that provide habitats and breeding grounds for a wide variety of organisms have been lost. According to investigations conducted in 1989, in the 13 years since the last survey, 6,403 hectares, or 3.2% of the existing seagrass bed area, had disappeared [2]. It was also found that in the areas of the Seto Inland Sea and the sea area that may be strongly affected by the inland seawater, the area of the seagrass and seaweed beds that disappeared was 20.8% of the national total

disappeared habitats. A detailed degradation of seagrass and seaweed habitat in western part of the Seto Inland Sea is shown in Table 1.1 [3].

Table 1.1 Seagrass and Seaweed habitat degradation

area	Habitat category	Area of seagrass and seaweed habitat (ha)		
		1989-1990	2017	
		4th Natural Environment Preservation Survey	Hearing	Satellite Image Analysis
Tokuyama Bay	seagrass	108.2	193.8	-
	seaweed	430.0	297.1	-
	total	538.2	490.9	294.2
Yamaguchi Bay	seagrass	0.0	286.2	-
	seaweed	230.8	173.3	-
	total	230.8	459.5	570.7
Koguchi	seagrass	0.0	0.0	-
	seaweed	404.0	368.2	-
	total	404.0	368.2	692.9
Kita-kyushu	seagrass	0.0	0.0	-
	seaweed	2005.0	2013.4	-
	total	2005.0	2013.4	934.3
Nakatsu	seagrass	0.0	516.1	-
	seaweed	1093.9	0.0	-
	total	1093.9	516.1	210.5
	seagrass	125.5	49.0	-

Kunisaki island	seaweed	54.0	128.1	-
	total	179.5	177.1	95.4
Saiki	seagrass	0.0	0.0	-
	seaweed	220.7	242.2	-
	total	220.7	242.2	115.0
Yawatahamashi	seagrass	0.0	0.0	-
	seaweed	181.5	156.6	-
	total	181.5	156.6	29.7

Taking precautions is urgent and important. According to the studies, only 5 to 10% of the world's seafloor is mapped [4]. Therefore, to assist effective seagrass and seaweed management and conservation, mapping is the first and urgent step.

1.1.2 Mapping seagrass and seaweed habitat

To map coastal ecosystems, recent advances in technology have increased available tools and accuracy. Traditionally, coastal habitat information can be obtained by field surveys with beam eco-sounders and side-scan sonar mounted on boats, submersible or remotely operated vehicles [5].

Satellite-based remote sensors, both free-accessed ones like Landsat and Sentinel and commercial ones such as IKONOS, WorldView, and Quickbird have been successfully used for benthic mapping with high to moderate resolutions [6]. Other than multispectral satellite imageries, hyperspectral satellite images and airborne remote sensing technologies are also available in the literature [7][8].

1.2 Technique for mapping

Machine learning algorithms have been popular in classifying benthic habitat[9]. This kind of methods, such as Random Forest and Support Vector Machines, takes ground truth data as input to train the classifier and predict the whole area with the support of spectral radiance or reflectance of satellite imagery.

Knudby [9] investigated the ability to use satellite images to classify coastal vegetation of several classification methods and found out that Random Forest outperformed other classification algorithms. The main classification targets are fringing reef, seagrass, and mangrove forest. And satellite images used in this study are Landsat scenes. After preprocessing interferences caused by the atmosphere and water column are removed and then 4 ensemble classification algorithms, random forest (RF), support vector machine (SVM), linear discriminant analysis (LDA), and penalized linear discriminant analysis (PLDA), are constructed and then assessed. The assessment was done by comparing overall accuracies that were produced using k-fold cross-validation with k=10. Among the four algorithms, RF yielded the highest 73.1% accuracy, while for SVM the number is 64.9%.

Roelfsema [6] utilized different sources of commercial satellite products for seagrass cover and species mapping. Three high-resolution satellite sensors are investigated in this study: Quickbird-2, IKONOS-2, and WorldView-2. The cover and percentage cover maps were created using a hierarchical Object-Based-Image-Analysis approach proposed by Blaschke.8 Mapping results were then assessed by accuracy. The accuracy of species composition ranged from 68% to 83%, with a median of 77%. In terms of overall accuracy, it showed a median of 52% accuracy. Combined with field data, satellite imagery showed an excellent ability to classify coverage and species composition.

Other than free-access satellite images like Landsat and high-resolution commercial alternatives like Quickbird, IKONOS, and Worldviews that have been researched, there is one new commercial product called PlanetScope that provides high-resolution with 1-day revisit time that shows potential in the field of coastal vegetation classification [10].

1.3 Literature review

Being one of the newest high-resolution products, PlanetScope provides 3-m resolution images on a daily basis [10]. Although not much, there are a few studies that use PlanetScope to classify seagrass habitat.

Wicaksono [11] studied the possibility of using PlanetScope to map benthic habitat in Indonesia. Random forest was chosen as the classification algorithm and applied on one image recorded on 18 July 2018. To train the model, training data were obtained by field survey conducted in the same period of satellite image acquisition time. One part of the training data set was used to train the classifier and the other part was used to assess the accuracy of classification. As a result, the overall accuracy of 60.6% and 78.6% were gained for the two different study sites. It mentioned that the accuracy was lower than other similar studies which applied random forest as the classifier, due to the reason of product limitation. This leaves room for improvement of utilizing PlanetScope to complete the same objective.

Ariasari and Wicaksono [12] made another attempt mapping both seagrass existence and species compositions. Random forest was also adopted this time, combined with field

surveys. The product used here was recorded on May 25, 2019. Kappa and overall accuracy were used for assessment. The highest kappa index was 0.54, and the highest overall accuracy is 72.09% in terms of benthic habitat mapping. This result did not change much in comparison with the former study. However, higher accuracy and kappa were achieved when mapping species composition, with a kappa value of 0.81 and accordingly overall accuracy value of 84.71%. It showed the potential of utilizing PlanetScope to map detailed information, however, the problem of relatively lower habitat mapping accuracy remains the same.

Munir and Wicaksono [13] changed the classification method and investigated the ability of PlanetScope again. Support vector machine was used in this study, and an overall accuracy of 73.98% was gained.

1.4 Objective

Although few studies tried to use PlanetScope to map seagrass habitat, the accuracy was not satisfied. However, by reviewing the methods used in those studies, improvements are expected to be achieved in two ways. Hence, this study aims to improve the accuracy of classification of seagrass and seaweed by PlanetScope and discuss how we can make the best use of this new resource. To be more specific, first, multi-temporal satellite images are used as input datasets to train the machine learning algorithm followed by a comparison with a single image input model. Second, two new features, water depth and distance to coastline are added into the classifier to improve the performance.

1.5 Significance and originality

By adding more features and data, improvement can be made and methods to maximize

the potential of PlanetScope imagery will be proposed and discussed.

Before this study, there are only very limited challenges that utilize this new satellite imagery and the results were not as promising as compared with other high-resolution commercial alternatives. Nevertheless, by proposing two ways to realize the full potential of PlanetScope, it provides opportunities and directions for researchers in the same or similar domain to consider using this new and high-quality satellite image product.

2 Materials and Methods

The current study involved classifying seagrass and seaweed habitat from satellite images using Random Forest (RF) in the Seto Inland Sea. Clear shallow water in Suo Nada was selected as the study area due to the wide distribution of its seagrass and seaweed beds. Random Forest is chosen as the classification method based on its ability to classify with high accuracy and functional capacity to handle noisy data.

2.1 Study site description

Suo Nada, located in the western part of the Seto Inland sea, has abundant tidal flats, seagrass and seaweeds. According to an investigation conducted in 2019 [3], there are 1925 ha of seagrass and seaweeds in Suo Nada, with 72% of them in Yamaguchi Pref. This study also aims to classify seagrass and seaweed beds in Suo Nada of Yamaguchi Pref, as shown in Figure 2.1 [14].

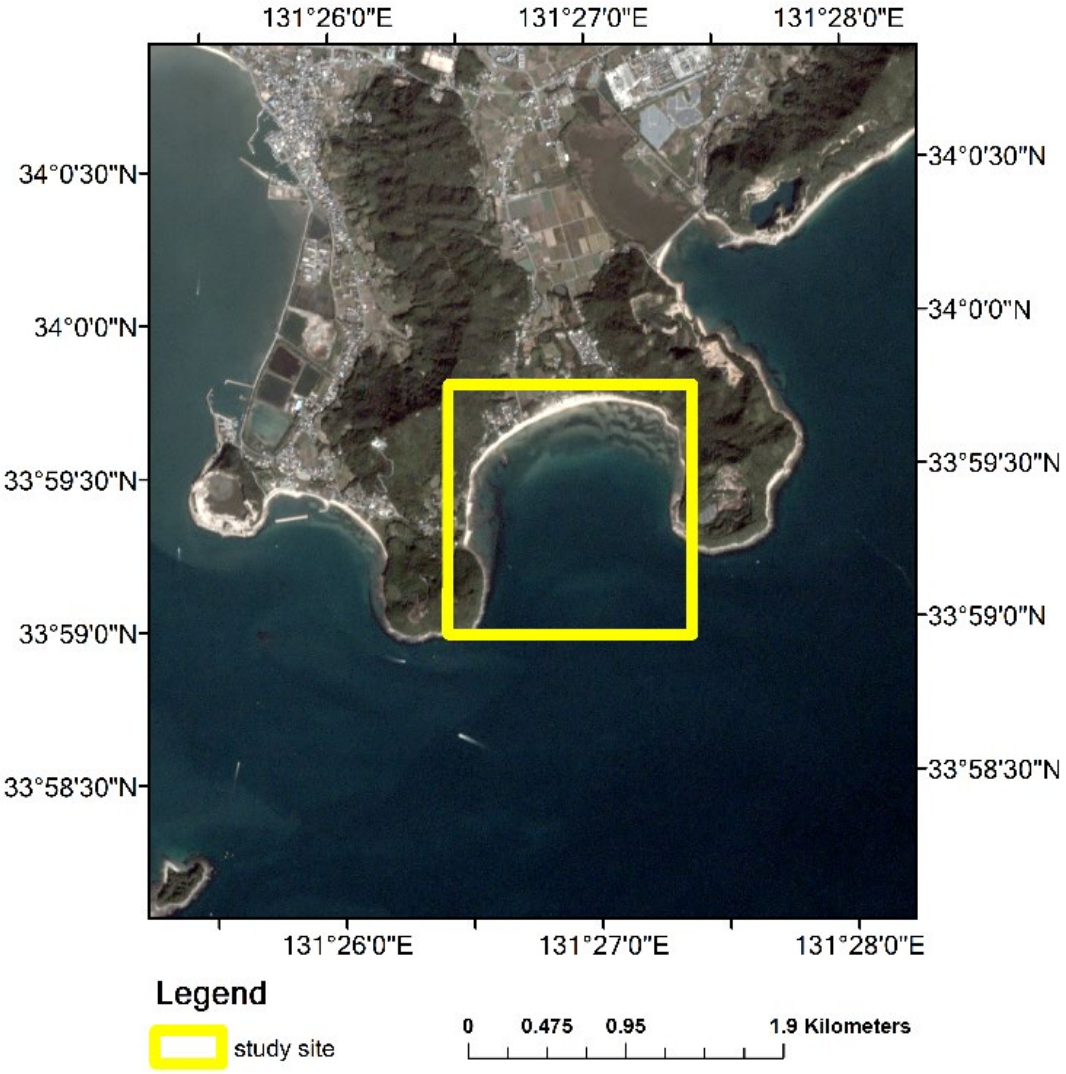


Fig. 2.1 Location of study site, Suo Nada, the Seto Inland Sea (Date of the satellite image: Oct 31, 2017) (Source: PlanetScope, Planet Application Program Interface: In Space for Life on Earth. San Francisco, CA. https://api.planet.com)

The study site is selected according to bathymetry and data availability. Theoretically, to conduct image classification through satellite imagery, the area to be classified must be optical shallow; hence the features are evident for the algorithm to work efficiently. Following the requirement, Suo nada and Hibiki nada have met the optical shallow water environment. However, only Hibiki nada is excluded as the study site since there is no

available fine-resolution bathymetry data of this area. To conclude, Suo nada is chosen as study site since it not only has the feature of shallow water where the seafloor is still adequately visible from the remote-sensing image because a 10-m mesh bathymetry data is available here.

2.2 Data

2.2.1 PlanetScope

PlanetScope has two strengths compared with other high-resolution commercial products with shorter revisit time and significantly bigger cover areas [15]. A comparison is shown in table 2.1.

Table 2.1 comparison between widely used commercial products

Name	Revisit frequency (day)	Resolution (m)	Capacity (million km^2 /day)
PlanetScope	1	3	340
WorldView-2	1.1 - 3.7	1.85	1
RapidEye	1 - 5.5	6.5	6
IKONOS	3	3.2	0.24
QuickBird	2 - 12	2.44	0.2

In this research, level 3B PlanetScope images recorded on October 27 and 31, November 2 and 5, 2017 were used, as shown in Table 2.2.

Table 2.2 four images used in this study

Image No.	Date	Recorded time (UTC)	Cloud percentage (%)	Satellite ID

1	October 27, 2017	01:14:28	16	0f34
2	October 31, 2017	02:07:25	0	0f21
3	November 2, 2017	02:07:23	2	1051
4	November 5, 2017	02:07:07	0	1051

PlanetScope is one of the newest high spatial resolution satellite imaging and can record an area of 150 million km2 per day [10]. Utilizing PlanetScope images for mapping coastal habitats has many advantages, especially with its 1-day revisit time strength, real-time information can be obtained and analyzed to investigate the impacts of extreme events like storm surges and other disasters in the coastal area. A detailed constellation overview of PlanetScope is given in Table 2.3[10].

Table2.3 constellation of PlanetScope

Mission characters	International space station orbit	Sun-synchronous orbit
Max/Min Latitude Coverage	± 52° (depending on season)	± 81.5° (depending on season)
Equator Crossing Time	Variable	9:30 – 11:30 am (local solar time)
Sensor	Three-band frame Imager or four-band frame Imgaer with a	

Type	split frame NIR filter			
Spectral Bands	Blue	Green	Red	NIR
	455 – 515 nm	500 –590 nm	590 – 670 nm	780 – 860 nm
Frame Size	20 km x 12 km (approximate)		24.6 x 16.4 km (approximate)	
Camera Dynamic Range	12-bit		12-bit	

Two kinds of products are provided officially [10]: At-sensor Radiance product and Surface Reflectance product. Initially, all PlanetScope satellite images are collected at a bit depth of 12 bits, and then radiometric corrections are applied during ground processing, which scales the 12-bit imagery to a 16-bit one. This scaling involves converting directly received Digital Numbers (DN) into at-sensor radiance, as is given by Eq. (1).

$$\mathrm{RAD(i)} = \mathrm{DN(i)} * \mathrm{radiometricScaleFactor(i)} \quad (1)$$

where radiometricScaleFactor(i) = 0.01.

By completing the radiometric correction, the product now represents the calibrated radiance and this is the At-sensor product.

Meanwhile, other than at-sensor radiance product, PlanetScope also provides surface reflectance product [10], called level 3B product. The one used in this study belongs to surface reflectance product that has been through radiometric correction and atmospheric

correction. Atmospheric correction aims to remove the interference of atmosphere and is processed by provider as follows. First, top of atmosphere reflectance is required by multiplying ReflectanceCoefficient provided by PlanetScope as is given by Eq. (2):

$$REF(i) = DN(i) * ReflectanceCoefficient(i) \quad (2)$$

Then, surface reflectance is determined from top of atmosphere (TOA) reflectance obtained from equation (2) by performing subsequent atmospheric correction conducted by combining the use of atmospheric models with the use of MODIS water vapor, ozone, and aerosol data. This preprocessing is also conducted by PlanetScope and the surface reflectance data that has reliable and consistent surface reflectance scenes and is ready for analysis for users.

2.2.2 Actual Distribution Map

In this study, the actual habitat distribution of seagrass and seaweeds derived from a large-scale investigation conducted by the Ministry of Environment was used as training data to train the machine learning classifier. This study's distribution data is named *Seagrass and Seaweed and tidal flat distribution map of the west part (GIS data)* and is part of the *Survey on the distribution of seaweed beds and tidal flats in the Seto Inland Sea* investigation and can be accessed via open database. This distribution was gained from a fusion method combining satellite images analysis with field survey validation.

The actual distribution is shown in Fig 2.2. [14][16]

Fig. 2.2 Distribution of seagrass and seaweed beds and tidal flat. Blue shows seagrass and seaweed beds and green is tidal flat. (Source: Created by processing the Survey on Distribution of Seagrass and Seaweed Beds and Tidal Flats in the Seto Inland Sea by Ministry of the Environment. http://www.env.go.jp/water/heisa/survey/result_setonaikai.html)

Where green stands for tidal flat, and blue stands for seagrass and seaweed habitat.

2.2.3 Water depth

Bathymetry data [17] with 10 m mesh water depth was used. The data is named *Tsunami fault model (5) Topographical data* and is provided by Study Group on the Nankai Trough Giant Earthquake Model organized by the Cabinet Office. It was obtained from an open-access website, geo-spatial data center and subsequently transferred from original '.dat' file to 10 mesh raster file follow information provided officially. Water depth data is formatted in the form that value of water points is positive and for land the value is negative. The datum of the bathymetry data is the Tokyo Peil (T.P.). The Planar Right Angle Coordinate System (JGD2000) was used as a projection method for data preparation.

To prepare the ".dat" data into usable raster data, two processes have been done. First, spatial information stored in description file was added to water depth ".dat" data. By doing this, each data point has its accordingly latitude, longitude and water depth. Second, the point data is transferred into 10 mesh raster layer data. This step enables extraction of water depth data to training points. These two steps were performed through Python and ArcGIS.

2.3 Method

2.3.1 Image standardization

c It is a common requirement for many machine learning estimators: they might behave badly if the individual features do not more or less look like standard normally distributed data[18]. In this study, standardization was performed through Python using Numpy package, as is given by Eq. (3)

$$z = (x - u)/s \tag{3}$$

where u is the mean of the surface reflectance of each band and s is the standard deviation.

2.3.2 Multispectral classification

Random forest classification is an ensemble classification method consisted by many decision trees [19]. The idea to combine multiple decision trees came from loss of generalization ability of a single decision tree model. Ho et al proposed a method[19] to construct tree-based classifiers, namely random forest, whose capacity can be arbitrarily expanded for increases in accuracy for both training and unseen data. The essence of the method is

to build multiple trees in randomly selected subspaces of the feature space and generalize their classification in complementary ways. By doing so, the combined classification produced by random forest, can be monotonically improved compared with results generated by decision tree model.

A diagram showing how random forest works is given in Fig 2.3. After a trainings set is fed into the classifier, different decision trees will be generated with each only consists random elements from the original dataset. Subsequently, each decision tree will construct their own decision process and generate a result. The ultimate classification result of a random forest is then a combination of results calculated by each decision trees within this forest.

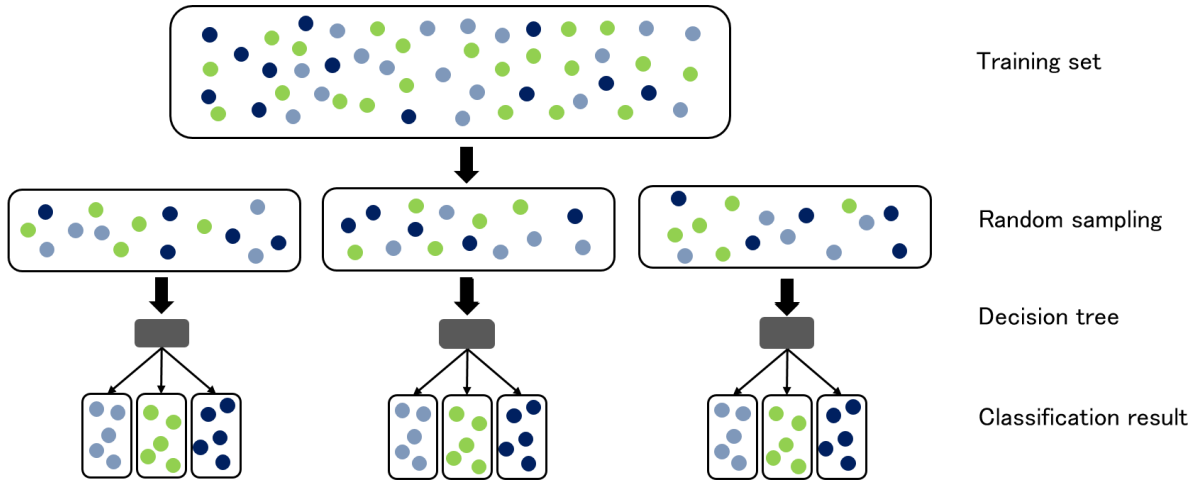


Fig. 2.3 Mechanism of random forest

The strength of this method comes from the high accuracy and good generalization performance, but it also lies in its ability to handle noisy data. It is considered crucial in dealing with PlanetScope dataset because one past study suggests that the signal-to-noise ratio of PlanetScope is low. Hence RF is chosen as the classification method in this study.

The algorithm was tuned in Python and ArcGIS to find the best setting (tree number and training size) for classification. For the classification, sklearn, Numpy, Pandas, matplotlib and seaborn packages are used. The workflow of this study is shown in Fig 2.4.

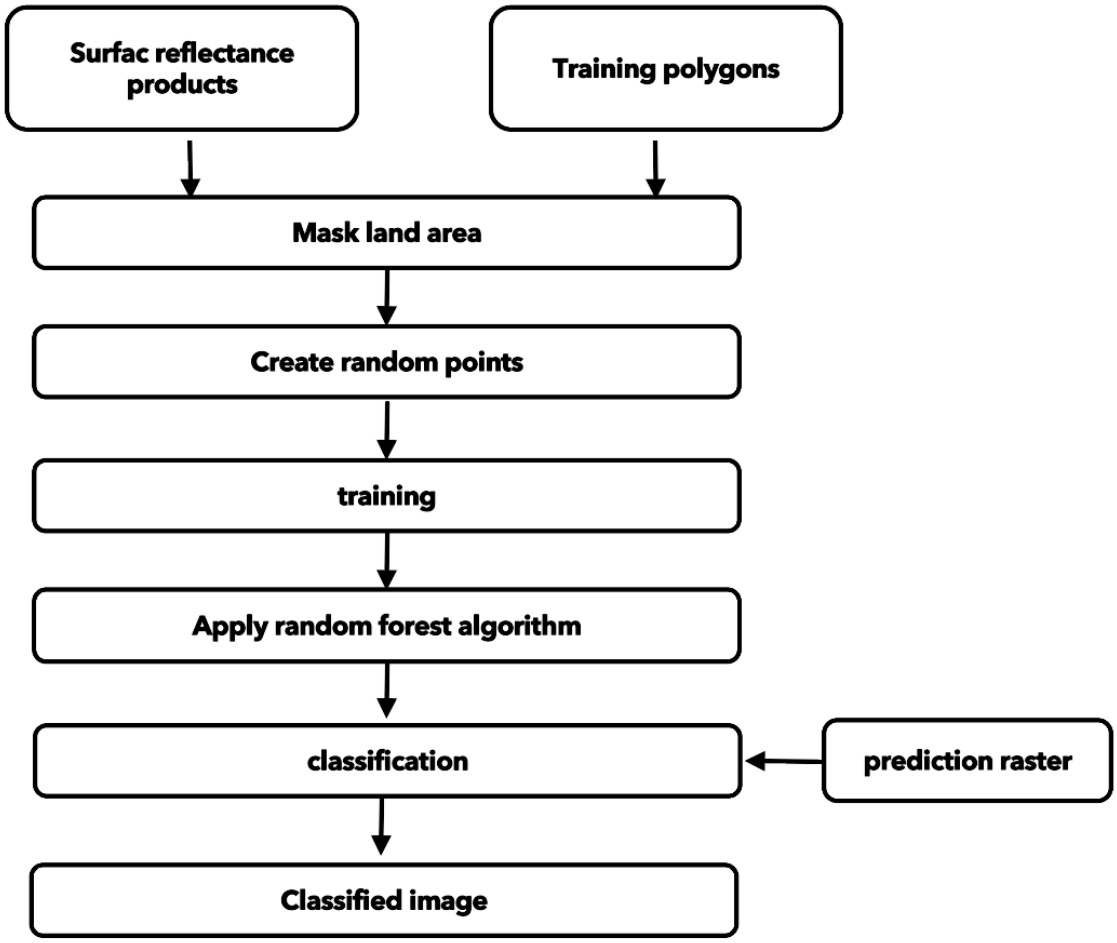


Fig. 2.4 Workflow of current study, including preprocess of satellite image and bathymetry data, training of random forest algorithm, followed by application of algorithm for classification

2.3.3 Mask land area

Land area was masked accordingly to coastline polygon data provided by ArcGIS Japan

and Geospatial Information Authority of Japan[20] that can be accessed and used free of charge. The product is called ‘National data for municipalities’ in the format of shapefile. Since the interest of this study is shallow water area, to avoid unnecessary computation time and improve classification accuracy, land area was masked out and target shallow water area was extracted, as shown in Fig 2.5. [14]

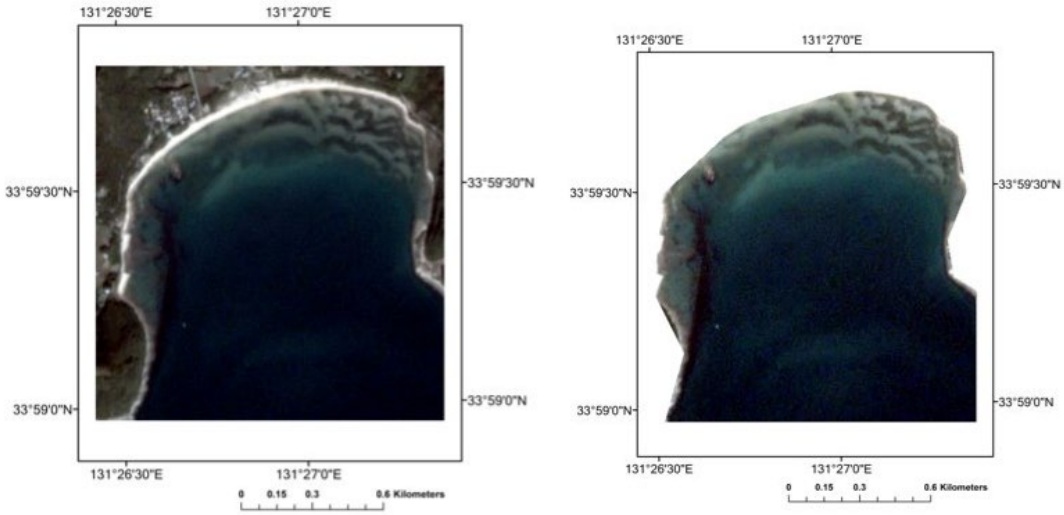


Fig. 2.5 Example of PlanetScope image before (left) and after (right) masking land area

2.3.4 Create training points

After surface reflectance product and training polygons were imported into ArcGIS, sampling is then required to create training set that, although only consists a small portion of the study site but can be presentative enough at the same time. To achieve this purpose, the ‘create random points’ function was applied to generate training points for each of the three classes: seagrass and seaweeds, tidal flats and water. When creating, each class should have the same or similar amount of data points to avoid bias that generated when each class has an imbalanced size of data [21].

For each class, 2000 points were created with consideration of model performance efficiency and stability while also left room for verification. To be more specific, to assure all three classes have similar or same amount of input data, downsizing the majority class is an effective way to achieve this purpose [21]. Hence in this study, size of water pixels has been reduced to align with the size of minority seagrass and seaweed pixels. As a result, in total 6000 points were generated for one satellite imagery as shown in Fig 2.6. [14]

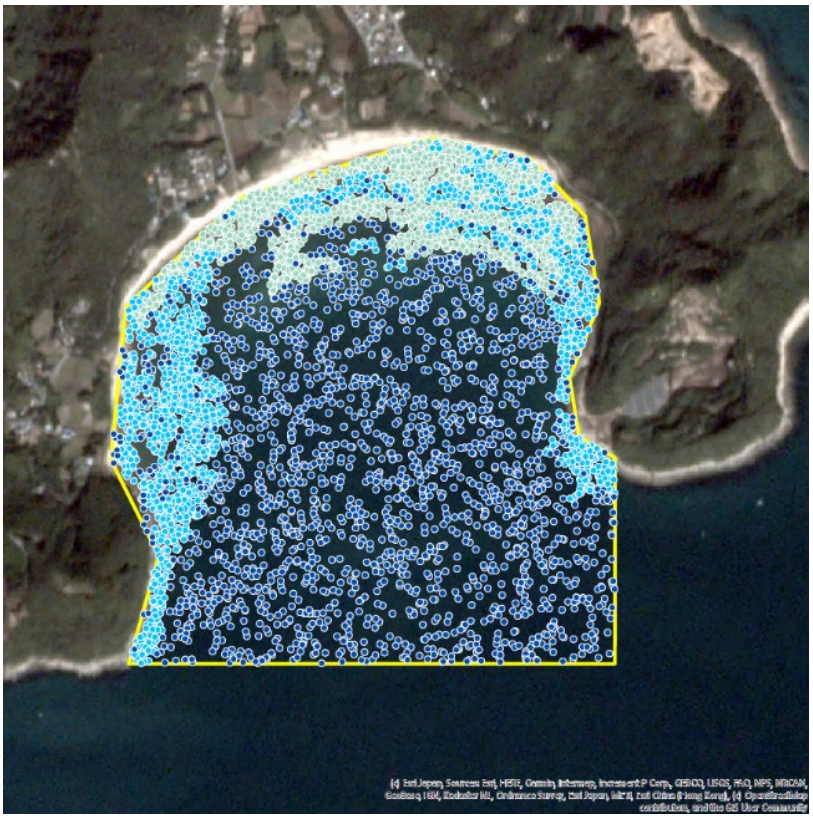


Fig. 2.6 Random sampling of three classes

2.3.5 Training

The classifier was then trained with six input features, including four standard scaled bands (blue, green, red, near-infrared), water depth and distance to coastline. Distance to coastline was calculated in ArcGIS using “near” function for each sampling point.

Two classifiers were trained and tuned, namely RF-1 and RF-2. RF-1 was trained with only 1 imagery, recorded on October 31, 2017. when tide level was low. RF-2 was trained with 4 different images input as mentioned in Table 2.3 before. Imageries were chosen based on the assumption that during short period of time, seagrass and seaweed distribution will not change significantly and thus could be seen as constant in this short amount of time.

Tuning process includes 2 parts, tree number adjustment and training size adjustment. Each setting is given in Table 2.4.

Table 2.4 Tuning category

Tree number	Training size for RF-1	Training size for RF-2
1	-	50
10	-	100
50	-	300
100	125	500
200	250	1000
500	375	1500
-	500	2000
-	750	3000
-	1250	5000
-	2000	8000
-	2500	10000
-	3250	13000

-	3750	15000
-	4500	18000

Since RF-1 used only one image as input while RF-2 used 4 images, the training size settings for RF-1 was one fourth that of RF-2. In terms of determining the upper limit of training size, consideration was paid to control the overall percentage of training set to whole pixels of study site. If the training size was created to be too large, then verification stage will be weakened. However, if training set was not large enough, the model performance will not reach the stable stage. Hence, when deciding training size, attention was paid to balance these two factors. To be more specific, in this study, the polygon area of each class has been taken into consideration to calculate the maximum possible distinctive number of data points. To get as many as distinctive data possible while consciously remain part of them for verification. As a result, the training size consists of around 2% of the whole study area.

2.3.6 Apply random forest algorithm

After parameter adjustment, the most efficient setting was found for each RF model. Among different parameter settings, the best one was chosen based on accuracy and precision of model performance and then was applied to four different dates of the same study area.

2.3.7 Classification

Classification was conducted for predict area. Based on the training model, it would perform the classification into three classes: seagrass and seaweeds, tidal flat and water. Confusion matrix, accuracy, precision and kappa index were calculated for evaluation of the performance of prediction.

2.4 Assessment method for model adjustment and classification result

2.4.1 Feature importance

Feature importance was calculated to assess the contribution of each parameter. It is computed as the (normalized) total reduction of the criterion brought by that feature, also known as the Gini importance [22]. In this study, feature importance was calculated in python with sklearn package.

Compared with previous studies [11][12][13], new features that represent environmental conditions were added in current study and an assessment to reveal whether these new features are important in model construction or not.

2.4.2 Confusion matrix

Confusion matrix is often used to calculate the accuracy of classification by comparing the number of correctly predicted objects and wrongfully predicted ones[23]. By definition, each entry in a confusion matrix comprises the number of observations belonging to class S/T/W but predicted to be in the class other than the true observation. The confusion matrix structure when a three-class classification is conducted is given in Table 2.5 below where S, T, W mean Seagrass and seaweed, Tidal flat, and Water.

Table 2.5 Structure of confusion matrix

		Predicted class			
		S	T	W	Total
Actual class	S	True Positive	False Negative (for S)	False Negative (for S)	X2
	T	False Positive (for S)	True Positive	False Negative (for T)	Y2

	W	False Positive (for S)	False Negative (for W)	True Positive	Z2
	To-tal	X1	Y1	Z1	

The term ‘True’ means the predicted class matches the actual class and on the contrary, the term ‘False’ means that the predicted category does not match the actual category. In terms of ‘Positive’ and ‘Negative’, they refer to the classification object, whether it is the target this task tends to identify or not. Hence, after understanding what these 4 terms stand for, it can be inferred that, whether a classification is satisfied enough can be determined by its confusion matrix. A better classification can be inferred in a manner that the numbers of True Positive and True Negative are dominant while the numbers of incorrectly predicted False Positive and False Negative are minor.

2.4.3 Cohen’s kappa index

Cohen’s kappa index [24] is a score that expresses the level of agreement between two annotators on a classification problem. In this case, it depicts the agreement between ground truth data and predicted type, as is given by Eq. (4):

$$\kappa = (p_o - p_e)/(1 - p_e) \tag{4}$$

where p_o is the observed agreement on the label assigned to any sample. It is calculated by the proportion of all classes on which predicted results and actual class agree). And p_e is the chance agreement that originally means when both annotators assign labels randomly when Cohen proposed. In classification evaluation, it means the portion that predicted class and actual class have same value by chance. Calculation of p_e is based on the number of observations counted in Table 2.5 and is given by Eq. (5):

$$p_e = X1 * X2 + Y1 * Y2 + Z1 * Z2 \quad (5)$$

The reason that cohen's kappa index was adopted as one of the evaluation methods in this study is due to its advantage that takes into account the possibility of the agreement by chance. This strength is considered more robust when assessing agreement between actual class and predicted class in categorical tasks.

This index was introduced in this study as an overall assessment of model performance that can deliver a straightforward comparison between two random forest classifiers.

2.4.4 Overall accuracy

After confusion matrix is calculated, overall accuracy (OA) can be obtained subsequently. It is calculated by summing the number of correctly classified values and then dividing by the total number of values, as can be expressed in Eq. (6). The correctly classified values, true positive and true negative, are located along the upper-left to lower-right diagonal of the confusion matrix.

$$OA = (N_{TP} + N_{TN}) \ / \ (N_{TP} + N_{TN} + N_{FP} + N_{FN}) \quad (6)$$

2.4.5 Precision

Only using OA as an assessment can sometimes be misleading since it reveals the accuracy of the whole classification task rather than one single target class that we are interested in. Hence precision [25] is introduced in parameter adjustment section of this study to give a more detailed look about one specific class, in this study, seagrass and seaweed. Precision is calculated by the number of true positive dividing by total predicted positive,

as can be expressed in Eq. (7):

$$Precision = N_{TP} / (N_{TP} + N_{FP}) \quad (7)$$

The point of utilizing precision for model construction is to provide information about false positive predicted results and then use this piece of information to filter parameter settings.

2.4.6 F1 score

Although precision provides detailed information about one specific class, it only measures one side of a coin – the cost of false positive side. However, there is another side: false negative side. In the task of seagrass classification, false negative means that the actual seagrass pixels cannot be identified correctly. This is the case we also want to avoid and the index reveals this ratio is Recall [25], calculation of which is given by Eq. (8).

$$Recall = N_{TP} / (N_{TP} + N_{FN}) \quad (8)$$

However, there is often a trade-off relationship between recall and precision. To balance precision and recall, F1 score[25] is introduced in this study for prediction assessment. In sklearn, the F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and the worst score at 0. The relative contribution of precision and recall to the F1 score are equal. It can be expressed as is given by Eq. (9):

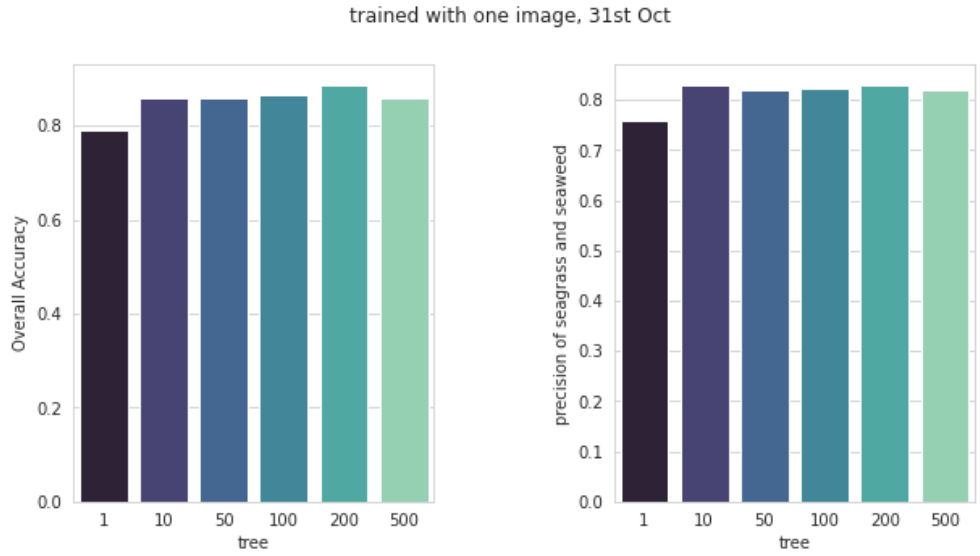
$$F1 = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (9)$$

3 Results

3.1 Model parameter adjustment

3.1.1 Tree number adjustment

Fig 3.1 shows the accuracy and precision of classifiers under different tree number settings. The best classifier is determined by taking both overall accuracy (OA) and precision of seagrass and seaweed into account.



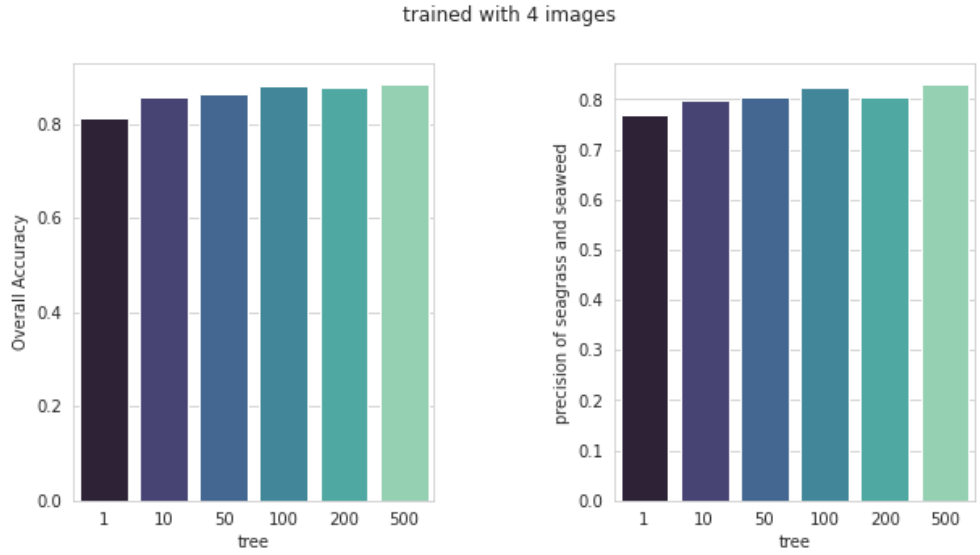


Fig 3.1 Tree number adjustment. Horizontal axis shows the number of decision trees in a random forest model and vertical axis shows the overall accuracy (left) and precision of seagrass and seaweed class (right).

As can be observed, compared with only 1 decision tree, all ensemble models have achieved higher OA and precision. For RF-1, trained with 1 imagery, the best tree number was 200 since it had yielded the highest OA and precision. For RF-2, the best tree number was 100. Hence, 200 and 100 were set to be the tree number of models.

3.1.2 Training size adjustment

Other than tree numbers, the best training size was also tested for both classifiers. Fig 3.2 shows model performance under different training size for RF-1 and RF-2.

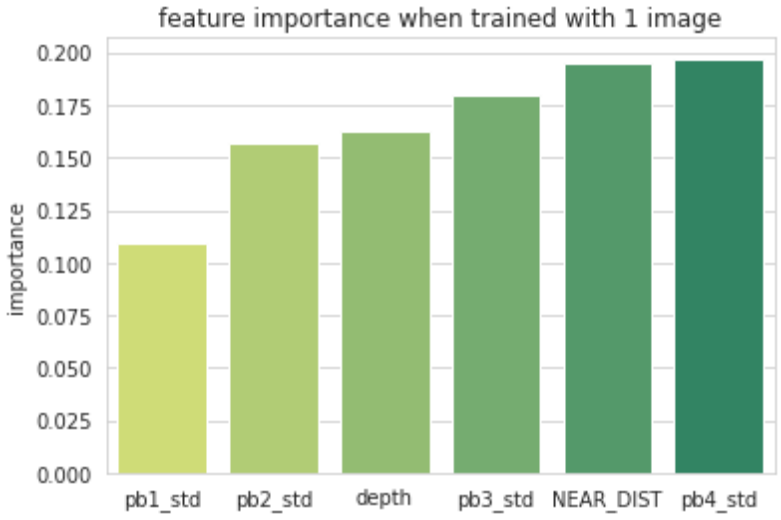
Fig. 3.2 Model performance under different training size . horizontal axis shows size of input training dataset and vertical axis shows overall accuracy (blue line) and precision of seagrass and seaweed class (orange lien)

It can be inferred that, when data size was too small, under 500, the model performance was not stable, meaning that even with a larger dataset, the performance will not increase accordingly. The threshold for this case is 500. It implies that, for a study area with a similar size of this study, at least 500 training point data are needed to achieve a reasonable result.

Second, when training size exceeded the least threshold of 500, accuracy and precision have increased following the increase of dataset. Follow this tendency, accuracy, and precision become stable when it reaches 3250 for RF-1 and 13000 for RF-2. After 3250 and 13000 training sampling points, there is no apparent significant increase in accuracy or precision. Hence these two values were chosen for model construction.

3.1.3 Feature importance

Feature importance was calculated for each RF to check the contribution of each input feature. The result is shown in Fig 3.3.



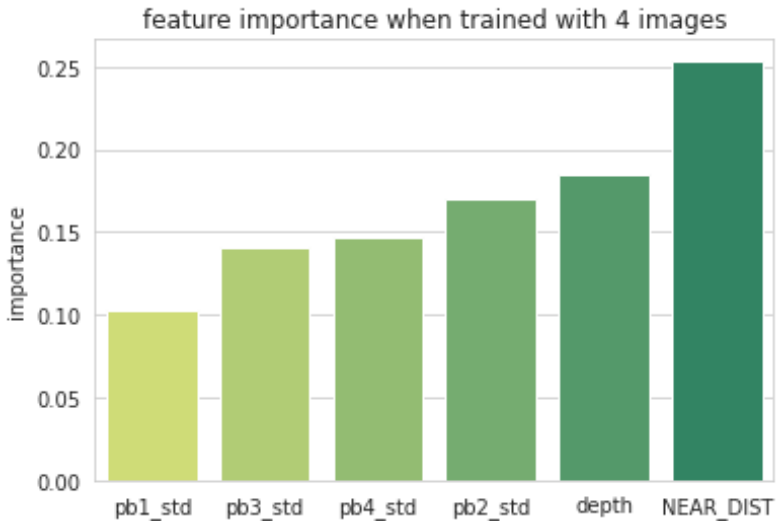


Fig. 3.3 Feature importance. pb1_std to pb4_std stands for standardization value of band 1 to band4, depth stands for bathymetry, NEAR_DIST stands for distance to coastline.

In this figure, pb1_std, pb2_std, pb3_std, and pb4_std stand for the standard scaled spectral bands – blue, green, red, and NIR. And NEAR_DIST and depth stand for distance to coastline and water depth, respectively.

In both models, distance to coastline and water depth have played essential roles in classification. This means that adding additional features other than the spectral band promotes improvement in model construction and performance. Other than the added features, among 4 visible bands, green band (pb2_std) and near-infrared band (pb4_std) contributed more, while the blue band (pb1_std) was relatively less useful in classification.

To conclude, despite some differences in the near-infrared band's contribution, these two models have similar feature importance in general.

3.2 Prediction assessment

3.2.1 Cohen's kappa index

Fig 3.4 shows kappa indexes of RF-1 and RF-2 when applied to four different satellite images. In this bar graph, cases 1 to 4 in the horizontal axis mean four different satellite images, they are October 27 and 31, November 2 and 5, 2017, respectively.

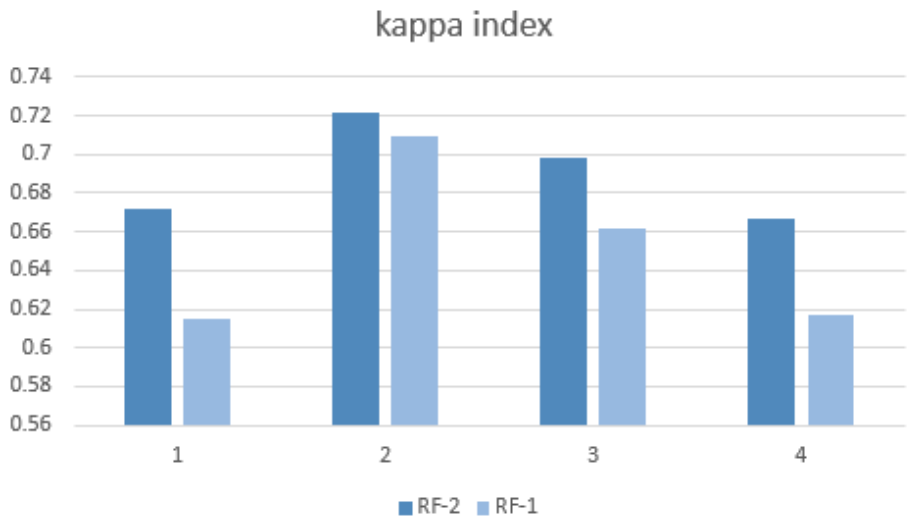


Fig. 3.4 Kappa index of two random forest classifiers. Dard blue bar shows results of RF-2 (the model that took four images as input) and light blue bar shows results of RF-1 (the mode that took only one image as input). Horizontal axis 1 to 4 mean application results on four different satellite images.

By comparing kappa between two different classifiers, we can come to two conclusions. First, under all four cases, RF-2 has slightly outperformed RF-1, by 0.01 to 0.06. Since kappa measure accuracy with consideration of the agreement by chance, it can be inferred that, with higher kappa value, RF-2 is more stable and generalizable than RF-1. Second, the extent of the difference is relatively different among the four cases. In terms of the possible reason behind this difference in kappa index, one possible reason is that, since RF-1 was trained using October 31, 2017 satellite image, the same one as case 2, hence

it is easier for RF-1 to perform better when applied to the same image. This leads to a smaller difference when applied RF-1 and RF-2 to case 2.

After comparison, then, focus only on RF-2. It can be inferred that, in case 2, which is the prediction on 31st Oct satellite image, kappa has reached the highest level. One possible reason for the difference in kappa index for four different satellite images could be the tide level when the image was recorded. Further explanations will be provided in the discussion section.

3.2.2 Overall accuracy

Fig 3.5 below shows the overall accuracy when applied RF-1 and RF-2 to four different images.

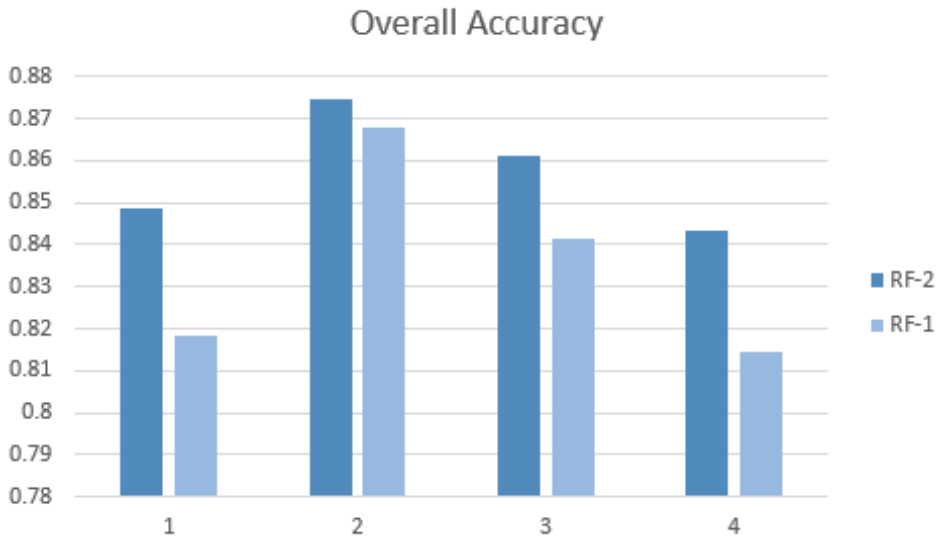


Fig. 3.5 Overall Accuracy. Dard blue bar shows results of RF-2 (the model that took four images as input) and light blue bar shows results of RF-1 (the mode that took only one image as input). Horizontal axis 1 to 4 mean application results on four different satellite images.

From this figure, it can be concluded that generally speaking Overall Accuracy results (OAs) are satisfying, with all of them over 80%, while it has reached the highest level under case 2, for both RF-1 and RF-2, with a value of 87.5% and 86.8% respectively.

Comparing OA and kappa index, the same tendency can be observed, where case 2 has yielded both the highest kappa value and the highest OA with RF-1 and RF-2. It is reasonable since both kappa and OA measure the overall performance of study instead of some specific class, and both provide a big picture for assessment.

3.2.3 F1 score

F1 scores of seagrass and seaweed class for RF-1 and RF-2 are shown in Fig 3.6.

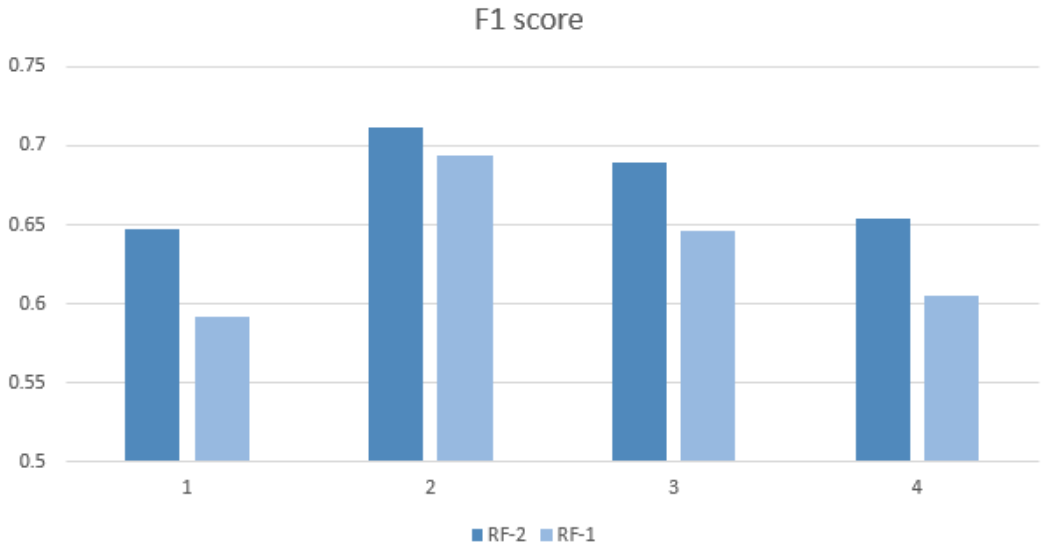


Fig. 3.6 F1 score. Dard blue bar shows results of RF-2 (the model that took four images as input) and light blue bar shows results of RF-1 (the mode that took only one image as input). Horizontal axis 1 to 4 mean application results on four different satellite images.

F1 score in this figure shows the balance between precision and recall for seagrass and

seaweed. In general, F1 scores are around or above 60%, with the highest reaching 71.2%. This can seem a satisfactory result. The overall tendency of f1 score for seagrass and seaweed is similar to the overall performance of both classifiers, where case 2 has yielded the highest overall and specific class accuracy.

3.2.4 Confusion matrix

Table 3.1 and Table 3.2 show the confusion matrix of RF-1 and RF-2 under case 2 (October 31, 2017).

Table 3.1 confusion matrix of RF-1, on October 31, 2017

		Predicted class		
		seagrass and seaweed	tidal flat	water
Actual class	seagrass and seaweed	17210	1638	132
	tidal flat	2885	21190	1015
	water	5300	13174	120044

Table 3.2 confusion matrix of RF-2 on October 31, 2017

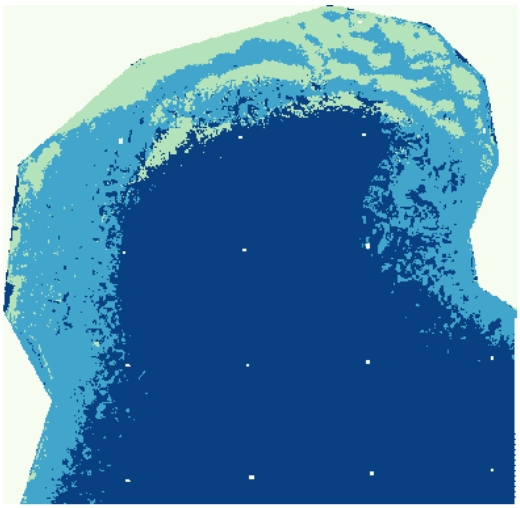
		Predicted class		
		seagrass and seaweed	tidal flat	water
Actual class	seagrass and seaweed	17422	1421	137
	tidal flat	3262	20837	991
	water	5852	11202	121464

From these two confusion matrices, we can conclude that, when RF-1 and RF-2 have both achieved their best performance, the confusion matrices look relatively similar. It indicates that trained with one best condition satellite image can give good results in this case.

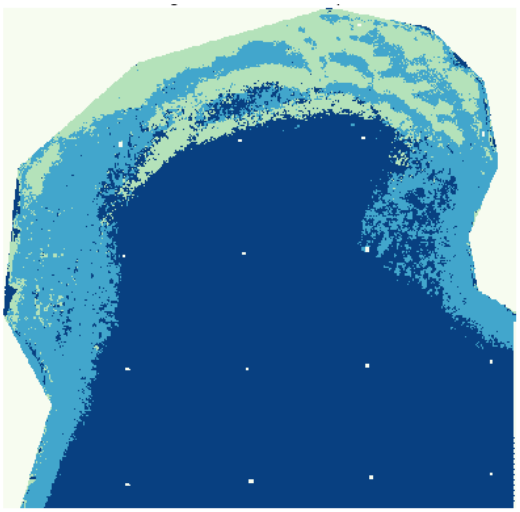
However, trained with 4 images still performed better if we focus on the number of true positive identifications: RF-1 and RF-2 have 17422 and 17210 true positive entries respectively. Other than the number of correctly predicted ones, incorrectly ones also need to be checked. In terms of false classification for seagrass and seaweed, RF-1 has less false positive entries, 8185 (2885 + 5300) while the number of RF-2 is 9114 (3262 + 5852). Another false classification is false negative, the number for RF-2 is 1558, smaller than that of RF-1, 1770.

3.3 Predicted map

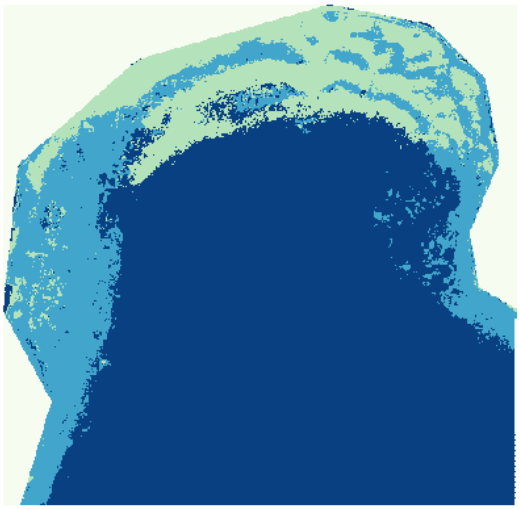
Fig 3.7 below shows the predicted map of RF-1 (left) and RF-2 (right) under case 1 to 4. Cases 1 to 4 mean different satellite images. Different colors mean different classes. Light blue is seagrass and seaweeds, light blue is tidal flat, and dark blue is water.08



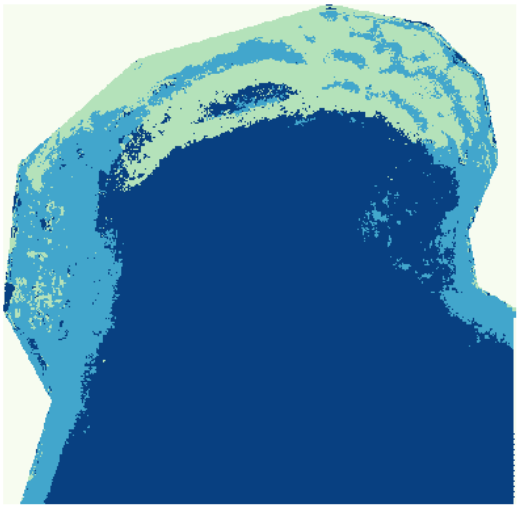
RF-1 on October 27, 2017



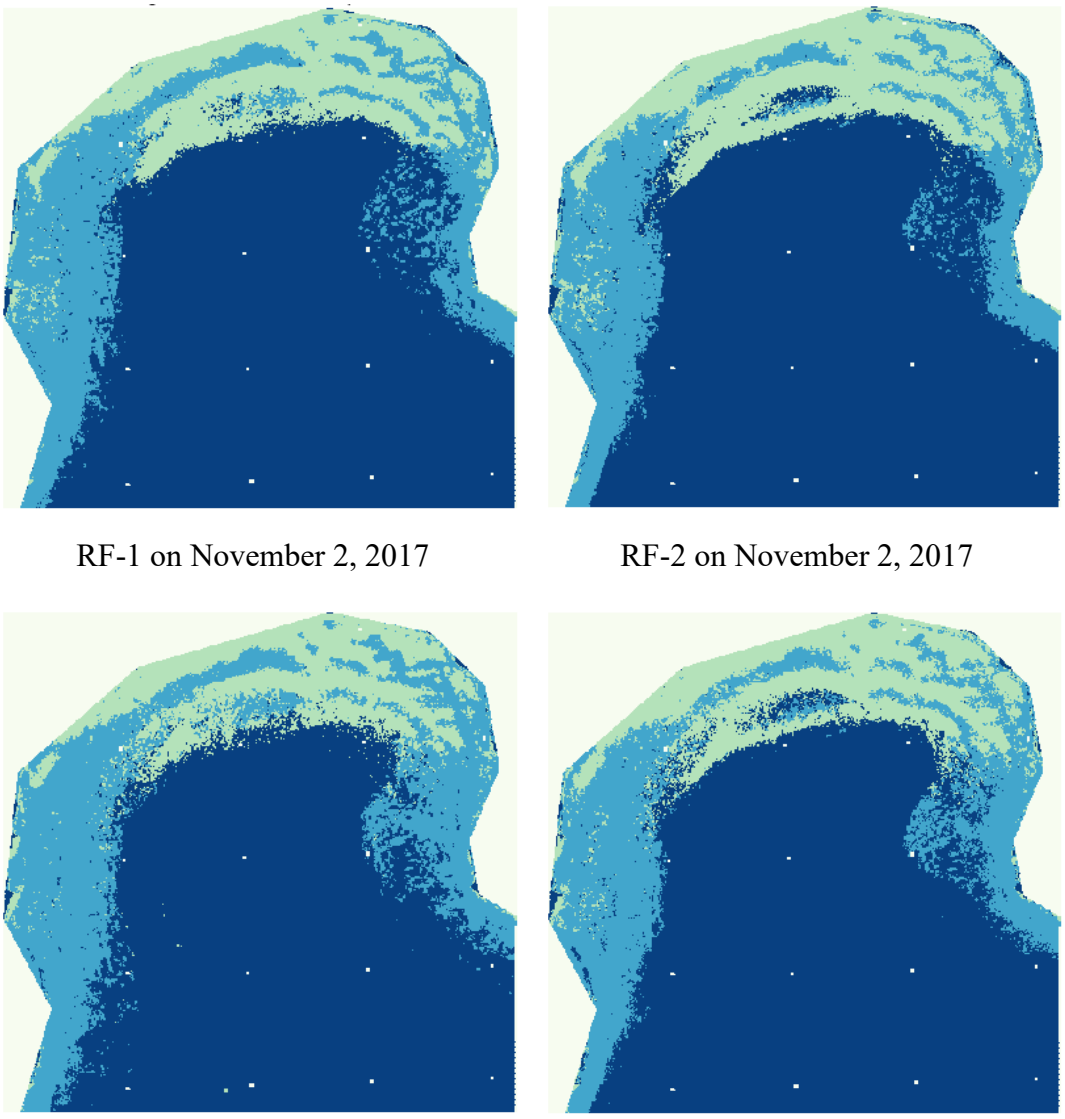
RF-2 on October 27, 2017



RF-1 on October 31, 2017



RF-2 on October 31, 2017



RF-1 on November 2, 2017　　RF-2 on November 2, 2017

RF-1 on November 5, 2017　　RF-2 on November 5, 2017

Fig. 3.7 Predicted map. Seagrass and seaweed beds are shown in blue, tidal flats are shown in green.

From the eight maps, we can come to three conclusions qualitatively.

First, plots on the left side are very similar to plots on the right side. This indicates that there is no significant difference between these two models.

Second, despite similar prediction result with a general look, by focusing on the middle right part of the study site, plots on the right side have a better ability to identify actual water pixels into water pixels, while plots on the left sides have less identification ability.

Last, the difference between the four applications is quite apparent. As can be observed by each row of this figure. The area and distribution of seagrass are relatively different among these four rows. Possible reasons will be discussed in the next section.

4 Discussion

4.1 Compare RF-1 and RF-2

From Cohen's kappa index, overall accuracy, f1 score, and confusion matrix provided in the previous section, we can quantitatively compare the two different classifiers.

Under all conditions, RF-2 has had better performance than RF-1. Considering that RF-2 took four images as input while RF-1 only took 1, theoretically speaking, the strength of RF-2 may come from both size and diversity of dataset. However, the accuracy would not improve significantly after a certain amount of size, even when input data size increased. Hence the improvement of performance yielded by RF-2 may come from the diversity of dataset. By saying diversity in this context, it refers to the environmental conditions of the time satellite images were recorded, e.g., tide level. To conclude, a more diversified dataset will help improve the performance of the classifier.

Besides comparison within this study, whether improvement has been made was checked by comparing it with another study using random forest to classify seagrass with PlanetScope imagery by Wicaksono [12]. The two OAs obtained in the previous study were 60.6% and 78.6% for two different study areas. However, by using multiple images as input data, OAs of this study have exceeded 80% under all cases, with the highest one reaching 87%. We are aware that a direct comparison of the current study and previous study will be less rationale since the study areas differ. Nevertheless, the current study has provided scientific evidence that using the same method and the same satellite source is promising to obtain high accuracy results.

4.2 Comparison among RF-2 application to four different images

In this part, a detailed comparison is made to explore more about the different prediction results on the four images. To define more clearly, cases 1 to 4 mean the applications of RF-1/RF-2 to images recorded on October 27 and 31, November 2 and 5, 2017. From the previous section, case 2 of RF-2 has yield the highest performance, both in the meaning of overall performance and specific seagrass and seaweed classification. Finding out the reason behind the difference in performance will provide valuable information for people interested in using PlanetScope to extract seagrass and seaweed habitat information. One apparent difference among the four cases is the tide level. Tide level for each image was referred to tide level charts provided by Japan Meteorological Agency. The tide levels for four cases are 175, 111, 148, 297 cm, respectively. A line graph is created, as shown in Fig 4.1 to show the correlation.

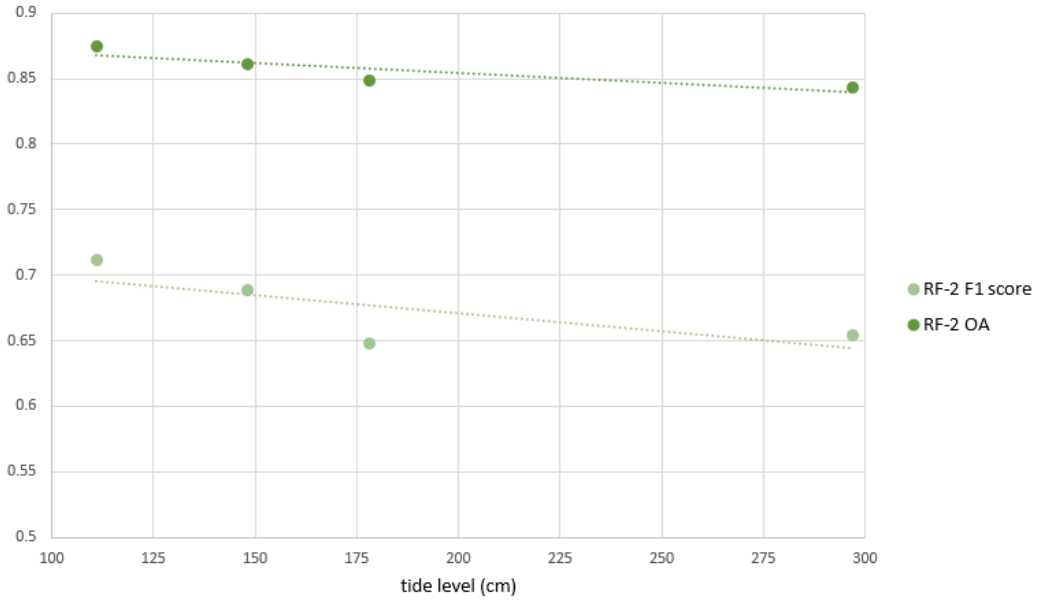


Fig. 4.1 correlation between tide level and OA, F1 score

It can be observed that, as tide level increases, the OA and F1 score tend to decrease. This

phenomenon may be because when the tide level is high, the percentage of submerged seagrass and seaweed increases; hence, it becomes difficult for the classifier to identify between underwater vegetation and water.

4.3 Observation on predict map and weak point of this model

Qualitatively, it can be observed from the eight predict maps provided in the previous session that the classifier easily mistakes the middle right part of the study area. Compared with ground truth distribution, the actual water pixels are likely to be predicted as seagrass and seaweed. To find out the possible reasons, we should first quantitatively evaluate the features of incorrectly identified pixels. Further analysis of RF-2 application on case 2 was conducted. From the confusion matrix provided in table 3.2, the most common classification mistake is predicted actual water pixel into seagrass and seaweed. The number of this false positive classification is 5852. Hence, to improve model performance, a detailed analysis is needed to first deal with this false categorization.

Three categories of pixels have been extracted: False Positive1 (TP1), True Positive1 (TP1), and True Positive2 (TP2). The meaning of these three classes is given in Table 4.1.

Table 4.1 meaning of FP1, TP1, TP2

Class name	Actual	Predicted
FP1	Water	Seagrass and seaweed
TP1	Water	Water
TP2	Seagrass and seaweed	Seagrass and seaweed

Fig 4.2 shows the distribution of FP1 pixels. From this figure, we can conclude directly

by observing that distances to coastline of these pixels are relatively small. This observation may lead to another discussion about the importance of distance to coastline feature.

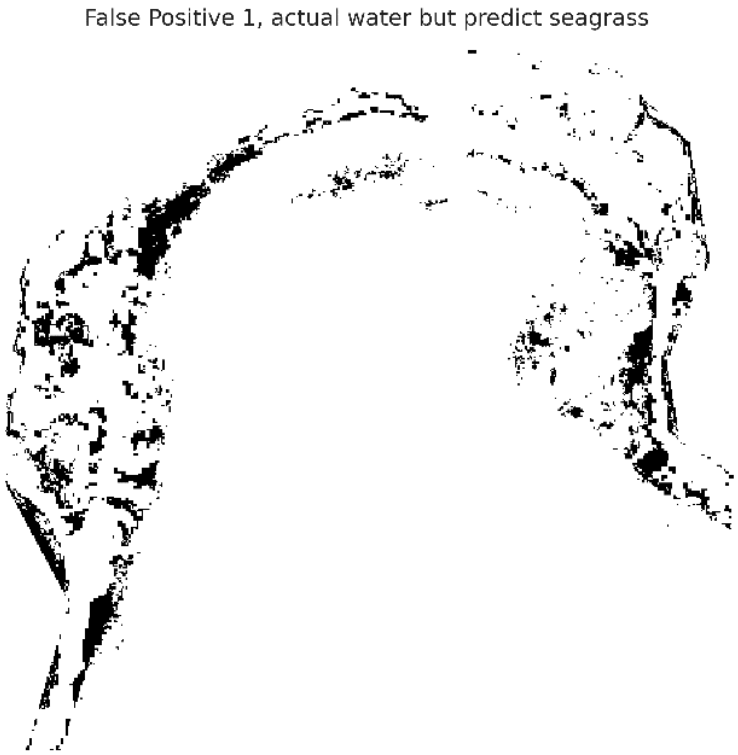
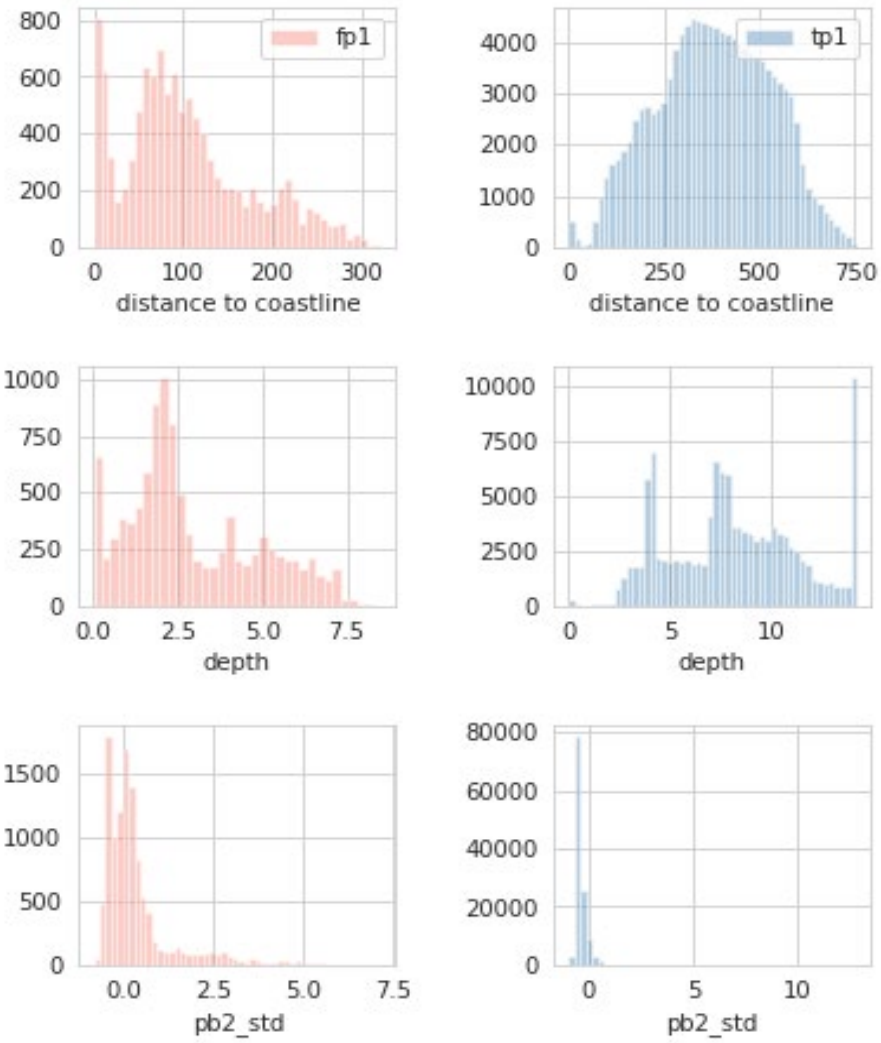


Fig. 4.2 Distribution of FP1 pixels. FP1 pixels are actual water pixels but were classified as seagrass and seaweed beds.

Having been revealed in Fig. 3.3, distance to coastline is a vital feature in classification. However, this can also be a weak point when the model relies heavily on it. As can be assumed, the average distance to coastline of water pixels should be more significant than that of seagrass and seaweed pixels. Hence, when the distance is not long enough, it gives the classifier an incentive to identify the pixel with short distance as seagrass and seaweed. This assumption can be verified by looking at the top 3 critical features of classification task among the three different categories, as shown in Fig. 4.3 Only three features, distance to coastline, water depth, and green band, were chosen for comparison since they

are the most important ones.



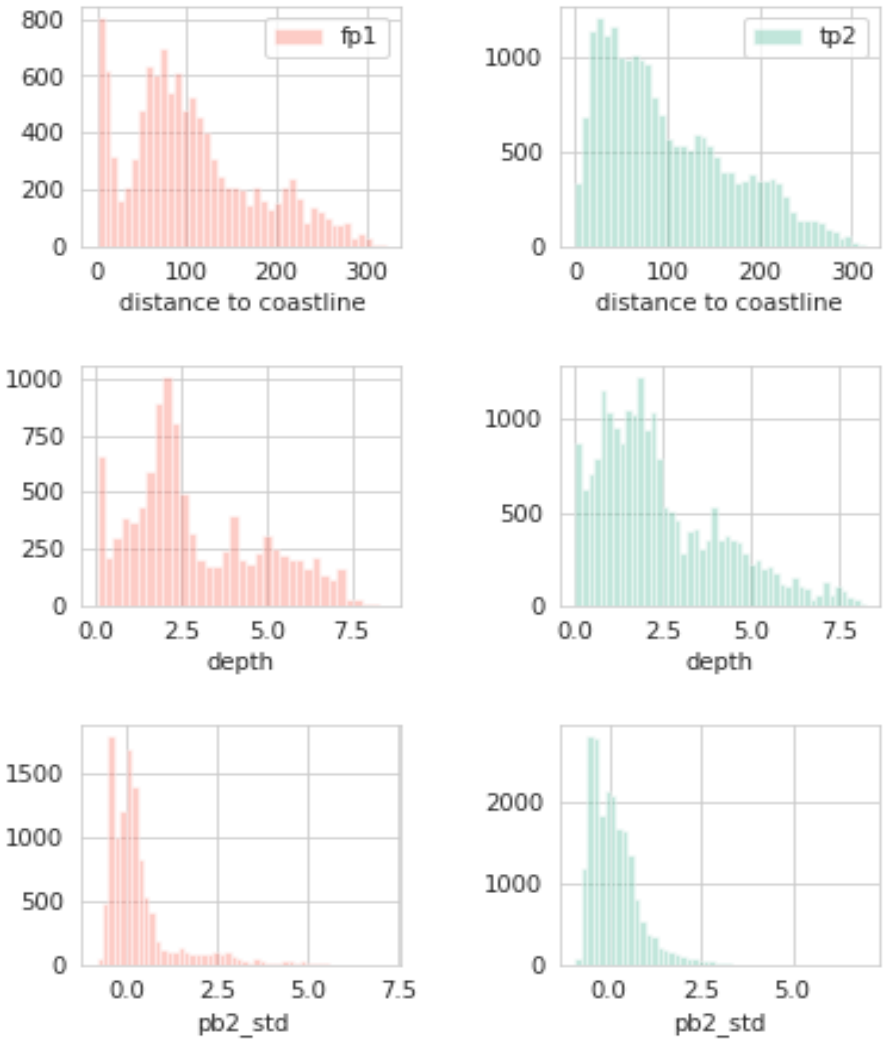


Fig. 4.3 Comparison between False Positive 1 and True Positive 1(upper panel, shown in red and blue color). Comparison between False Positive 1 and True Positive 2 (lower panel, shown in red and green color)

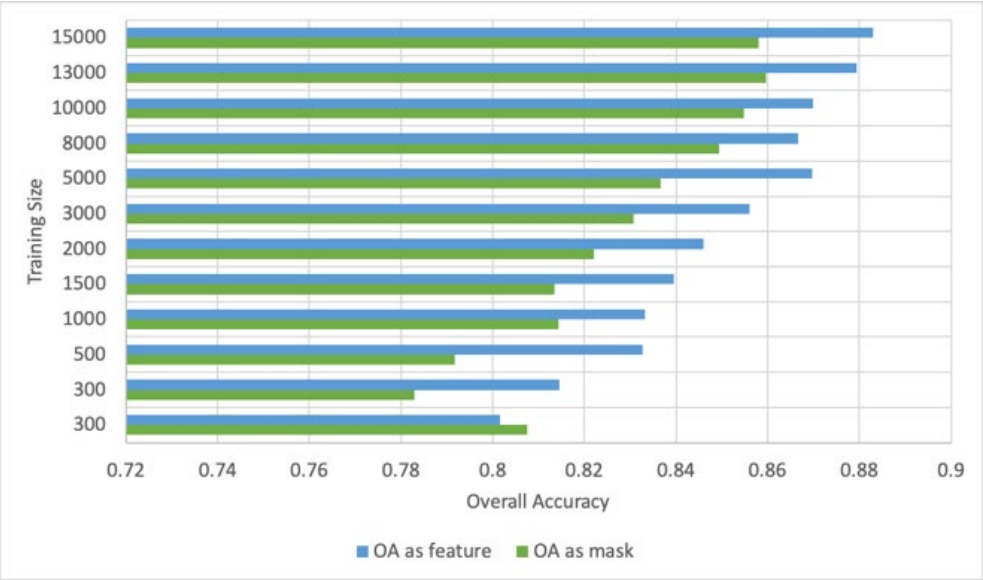
Each color stands for one category, where red is FP1, blue is TP1, and green is TP2. From observation, distance to coastline of FP1 distributes between 0 to 300m, with the majority ranging between 50 and 200 m. However, the value for TP1, water pixels, is between 0 to 750m, with a majority range of 50 to 600m. This confirms the assumption proposed above that distances of water pixels are relatively large. When the pixel with relatively

small distance, it is reasonable for the classifier not to treat it as water.

Other than TP1, a comparison between FP1 and TP2 has been made. By observing the six subplots in Fig. 4.3, we can conclude that FP1 and TP2 have very similar distributions. This indicates that, with similar feature characteristics, the classifier is likely to identify these two as the same class – seagrass and seaweed. However, in actual, they are not the same class: red in actual is water and green in actual is seagrass and seaweeds.

In order to solve this false positive classification weak point, three strategies can be adopted. First, more powerful features should be added to this classifier, to balance the weight of distance to coastline and make the classifier less reliable on one feature. Second, instead of using distance as a feature, an alternative usage is to apply it as a mask that works in a similar way to the land area mask. Last, by keeping the current model setting, we can improve the performance by using the best condition image – case 2 with the lowest tide level among the four images.

The first solution, finding other valuable features, unfortunately, is not solved in the current study. The second solution that applied a mask to the study site also failed to deliver a satisfactory performance compared with the original setting expressed in the methodology section, as shown in Fig. 4.4, Specifically, a mask was created with the threshold of maximum seagrass and seaweed pixel distance plus one standard deviation, meaning that pixels with larger distances than this threshold were masked out in this analysis.



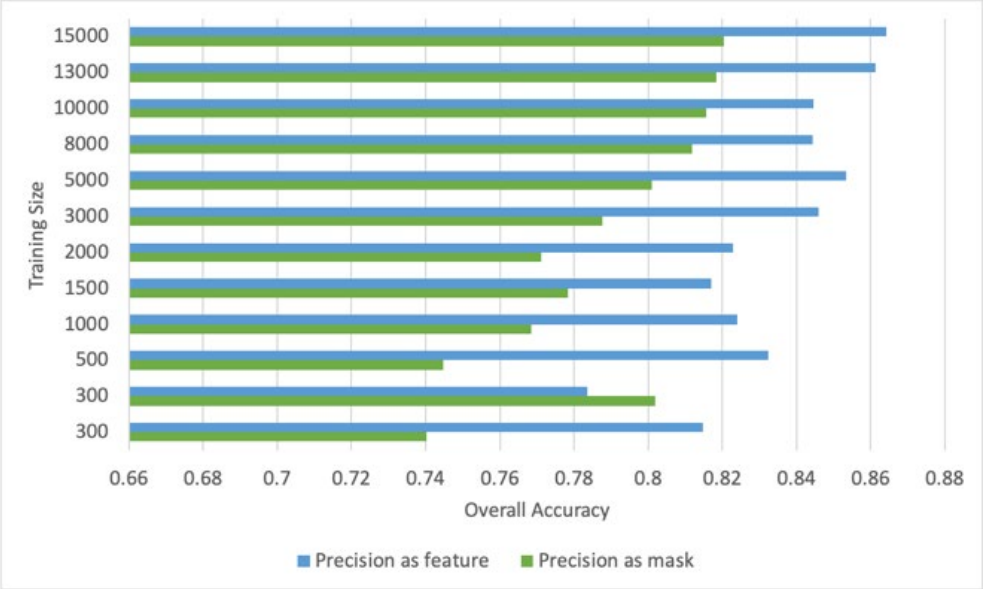


Fig. 4.4 Comparison of OA (upper panel) and precision of seagrass and seaweed (lower panel)

In these two figures, green bars stand for results of solution 2, which took distance to coastline as a mask. As a comparison, blue bars show the results of original model setting that took distance to coastline as a feature. By observing, blue bars are higher than green

bar in most situations in the sense of general performance of specific class identification performance. We can conclude that, it is better to take distance to coastline as a feature rather than a mask.

However, we find evidence to support the last solution – using better images does help improve performance. This can be observed by all the qualitative and quantitative analyses provided in section 3 (Fig 3.4 to 3.7). When applied the same RF-1 or RF-2 to four different images, the best condition image has yielded the highest performance. A possible reason behind this may be that, with the lowest tide level, the interference is reduced to least, and efficiency of spectral bands reach their highest level.

5 Conclusion

In terms of the overall objective of this study which is making improvements of PlanetScope imagery classification based on previous studies, we can conclude that the improvements have been achieved since the overall accuracy and kappa index of seagrass and seaweed beds classification have been improved significantly compared with previous researches of the same purpose.

To be more specific, the improvements have been made in two ways, by adding new features and adding more diversified data to train the model. And these two methods have been proved to be effective in improving classification performance. In terms of new features, environmental features that related closely to seagrass and seaweed distribution, such as water depth and distance to coastline, should be considered as input features since they provide valuable information compared with using only spectral band. In terms of adding more diversified data as input data, using multiple imageries increased model performance compared with using only one best condition imagery, and we assume that it is because multiple imageries enhanced generalization ability of the model by increased data amount and data diversity.

To summarize, it is concluded that from the results of this study, PlanetScope imagery has proved to be suitable for seagrass and seaweed beds classification and improvements have been made to realize more of its potential.

References

[1] G. Nellemann, C., Corcoran, E., Duarte, C. M., Valdés, L., De Young, C., Fonseca, L., Grimsditch, *Blue Carbon. A Rapid Response Assessment.* 2009.

[2] E. A. and M. P. C. of J. Nature Conservation Bureau, "The Report of the Marine Biotic Environment Survey in the 4th National Survey on the Natural Environment Vol.2 Algal and Sea-Grass Beds," 1989.

[3] M. of the Environment, "Survey on Distribution of Seagrass and Seaweed Beds and Tidal Flats in the Seto Inland Sea (Western Sea)," pp. 0–19, 2018.

[4] D. Wright and W. Heyman, "Introduction to the special issue: Marine and Coastal GIS for Geomorphology, habitat mapping, and marine reserves," *Mar. Geod.*, vol. 31, no. 4, pp. 223–230, 2008, doi: 10.1080/01490410802466306.

[5] M. U. Gumusay, T. Bakirman, I. Tuney Kizilkaya, and N. O. Aykut, "A review of seagrass detection, mapping and monitoring applications using acoustic systems," *Eur. J. Remote Sens.*, vol. 52, no. 1, pp. 1–29, 2019, doi: 10.1080/22797254.2018.1544838.

[6] C. M. Roelfsema *et al.*, "Multi-temporal mapping of seagrass cover, species and biomass: A semi-automated object based image analysis approach," *Remote Sens. Environ.*, vol. 150, pp. 172–187, 2014, doi: 10.1016/j.rse.2014.05.001.

[7] M. Valle *et al.*, "Mapping estuarine habitats using airborne hyperspectral imagery, with special focus on seagrass meadows," *Estuar. Coast. Shelf Sci.*, vol. 164, pp. 433–442, 2015, doi: 10.1016/j.ecss.2015.07.034.

[8] E. Peneva, J. A. Griffith, and G. A. Carter, "Seagrass Mapping in the Northern Gulf of Mexico using Airborne Hyperspectral Imagery: A Comparison of Classification Methods," *J. Coast. Res.*, vol. 244, no. 4, pp. 850–856, 2008, doi: 10.2112/06-0764.1.

[9] A. Knudby *et al.*, "Using multiple Landsat scenes in an ensemble classifier reduces classification error in a stable nearshore environment," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 28, no. 1, pp. 90–101, 2014, doi: 10.1016/j.jag.2013.11.015.

[10] Planet, "Planet Imagery : Product Spesification," *Planet*, no. January, p. 96, 2019, [Online]. Available: https://www.planet.com/products/planet-imagery/.

[11] P. Wicaksono and W. Lazuardi, "Random Forest Classification Scenarios for Benthic Habitat Mapping using Planetscope Image," *Int. Geosci. Remote Sens. Symp.*, vol. 167, pp. 8245–8248, 2019, doi: 10.1109/IGARSS.2019.8899825.

[12] A. Ariasari, . Hartono, and P. Wicaksono, "Random forest classification and regression for seagrass mapping using PlanetScope image in Labuan Bajo, East Nusa Tenggara," no. December 2019, p. 77, 2019, doi: 10.1117/12.2541718.

[13] M. Munir and P. Wicaksono, "Support vector machine for seagrass percent cover

mapping using PlanetScope image in Labuan Bajo, East Nusa Tenggara," no. December 2019, p. 112, 2019, doi: 10.1117/12.2541849.

[14] P. T. (2017)., "Planet Application Program Interface: In Space for Life on Earth. San Francisco, CA. https://api.planet.com."

[15] EARTHDATA, "Commercial Datasets," 2020. https://earthdata.nasa.gov/esds/small-satellite-data-buy-program/commercial-datasets.

[16] "Created by processing the Survey on Distribution of Seagrass and Seaweed Beds and Tidal Flats in the Seto Inland Sea (Ministry of the Environment)." http://www.env.go.jp/water/heisa/survey/result_setonaikai.html.

[17] C. O. Study Group on the Nankai Trough Giant Earthquake Model, "Tsunami fault model (5) Topographical data," 2019. https://www.geospatial.jp/ckan/dataset/1205 (accessed Jul. 06, 2020).

[18] "sklearn.preprocessing.StandardScaler — scikit-learn 0.23.1 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html (accessed Jul. 06, 2020).

[19] Tin Kam Ho, "Random Decision Forests," *Proc. 3rd Int. Conf. Doc. Anal. Recognit.*, pp. 278–282, 1995, doi: 10.1109/ICDAR.1995.598994.

[20] G. I. A. of J. Esri Japan, "National data for municipalities and basic information on national land," *2020*. https://www.esrij.com/products/japan-shp/ (accessed Jul. 06, 2020).

[21] A. Fernández *et al.*, "Learning from Imbalanced Data Streams," *Learn. from Imbalanced Data Sets*, pp. 279–303, 2018, doi: 10.1007/978-3-319-98074-4_11.

[22] "sklearn.tree.DecisionTreeRegressor — scikit-learn 0.23.1 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html (accessed Jul. 06, 2020).

[23] S. V. Stehman, "Selecting and interpreting measures of thematic classification accuracy," *Remote Sens. Environ.*, vol. 62, no. 1, pp. 77–89, 1997, doi: 10.1016/S0034-4257(97)00083-7.

[24] J. Cohen, "Cohen, J. (1960). A coefficient of agreement for nomial scales. Educational and Psychological Measurement, 20(1), 37–46. doi:10.1177/001316446002000104A coefficient of agreement for nomial scales," *Educ. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, 1960, doi: 10.1177/001316446002000104.

[25] D. M. W. Powers and Ailab, "Evaluation : From Precision , Recall and F-Factor to ROC , Informedness , Markedness & Correlation," *J. Mach. Learn. Technol.*, vol. 2, no. December, pp. 37–63, 2007, [Online]. Available: http://www.bioinfo.in/contents.php?id=51.

References of Python code

1 Reading:

1.1 Read the file prepared in ArcGIS into Python

```
moba = gpd.read_file('/content/drive/Shared drives/Qin
Xiaoyu 2017.10/west one-study-site/train whole
study/mstudy.shp')
moba = moba[moba['depth'] > 0]
print(moba.shape)
moba['labels'] = 'moba'

water = gpd.read_file("/content/drive/Shared drives/Qin
Xiaoyu 2017.10/west one-study-site/train whole
study/wstudy.shp")
water = water[water['depth'] > 0]
water['labels'] = 'water'

higata = gpd.read_file("/content/drive/Shared drives/Qin
Xiaoyu 2017.10/west one-study-site/train whole
study/hstudy.shp")
higata = higata[higata['depth'] > 0]
higata['labels'] = 'higata'


# break down and concat
moba.head()
cols_name = ['NEAR_DIST',  'depth', 'b1', 'b2', 'b3', 'b4']
cols1 = ['NEAR_DIST',  'depth', 'b1_p1027', 'b2_p1027',
'b3_p1027', 'b4_p1027']
cols2 = ['NEAR_DIST', 'depth', 'b1_p1031', 'b2_p1031',
'b3_p1031', 'b4_p1031']
cols3 = ['NEAR_DIST', 'depth', 'b1_p1102', 'b2_p1102',
'b3_p1102', 'b4_p1102']
cols4 = ['NEAR_DIST', 'depth', 'b1_p1105', 'b2_p1105',
'b3_p1105', 'b4_p1105']
```

```
# break down 1
moba1 = moba[cols1]
moba1.columns = cols_name
moba1['labels'] = 'moba'
water1 = water[cols1]
water1.columns = cols_name
water1['labels'] = 'water'
higata1 = higata[cols1]
higata1.columns = cols_name
higata1['labels'] = 'higata'

west1 = pd.concat([moba1, water1, higata1])

#scale. 2017-10-27
orig_planet = rasterio.open("/content/drive/Shared
drives/Qin Xiaoyu 2017.10/west one-study-site/pred whole
study/whole_1027.tif")
bband = orig_planet.read(1).ravel()
gband = orig_planet.read(2).ravel()
rband = orig_planet.read(3).ravel()
nirband = orig_planet.read(4).ravel()

mean1 = np.mean(bband[bband!=65535])
std1 = np.std(bband[bband!=65535])
print('mean and std of blue band is:', mean1, std1)
mean2 = np.mean(gband[gband!=65535])
std2 = np.std(gband[gband!=65535])
print('mean and std of green band is:', mean2, std2)
mean3 = np.mean(rband[rband!=65535])
std3 = np.std(rband[rband!=65535])
print('mean and std of red band is:', mean3, std3)
mean4 = np.mean(nirband[nirband!=65535])
std4 = np.std(nirband[nirband!=65535])
print('mean and std of nir band is:', mean4, std4)
```

1.2 standardization

```
# The standard score of a sample x is calculated as: z = (x
- u) / s
```

```
west1['pb1_std'] = west1['b1'].apply(lambda x : (x -
mean1)/std1)
west1['pb2_std'] = west1['b2'].apply(lambda x : (x -
mean2)/std2)
west1['pb3_std'] = west1['b3'].apply(lambda x : (x -
mean3)/std3)
west1['pb4_std'] = west1['b4'].apply(lambda x : (x -
mean4)/std4)
```

1.3 concat prepared 4 images into 1 dataframe

```
west_teach = pd.concat([west1, west2, west3, west4],
axis=0)
west_teach = west_teach.drop(columns = ['b1', 'b2', 'b3',
'b4'])
```

2 preparation for training classifier

2.1 create input feature

```
labels = np.array(west_teach['labels'])
features= west_teach.drop(['labels'], axis = 1)
feature_list = list(features.columns)
features = np.array(features)
```

2.2 parameter adjustment : tree number

```
test_oa = []
moba_prec = []
higata_prec = []
water_prec = []
sns.set_style('whitegrid')
sns.set_palette('Set3_r')

# x tress
from sklearn.model_selection import train_test_split
train_features, test_features, train_labels, test_labels =
train_test_split(features, labels, test_size = 0.1)
```

```
print('Training Features Shape:', train_features.shape)
print('Testing Features Shape:', test_features.shape)

# train
classifier = RandomForestClassifier(n_estimators = x)
classifier.fit(train_features, train_labels)

# test
west_pred = classifier.predict(test_features)

# accuracy, feature_importance and confusion matrix
print('accuracy is:',accuracy_score(test_la-
bels,west_pred),'\n')
test_oa.append(accuracy_score(test_labels,west_pred))

print("confusion matrix:",'\n',confusion_matrix(test_la-
bels, west_pred, labels=['moba','higata','water']),'\n')

from sklearn.metrics import precision_score
moba_prec.append(precision_score(test_labels, west_pred,
labels=['moba'], average=None)[0])
higata_prec.append(precision_score(test_labels, west_pred,
labels=['higata'], average=None)[0])
water_prec.append(precision_score(test_labels, west_pred,
labels=['water'], average=None)[0])


palette = sns.color_palette("mako")
sns.set_palette(palette=palette)

tree = pd.DataFrame([1,10,50,100,200,500]) # tree number
settings of this study
test_eval = pd.concat([tree, pd.DataFrame(test_oa),pd.Data-
Frame(moba_prec),pd.DataFrame(higata_prec),pd.DataFrame(wa-
ter_prec)], axis=1)
test_eval.columns =
['tree','oa','prec_m','prec_h','prec_w']
print(test_eval.head(),'\n')
test_eval.to_csv("tree_eval.csv")
```

```
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10,5))
plt.subplots_adjust(wspace=0.5)
sns.barplot(x='tree', y ='oa',data=test_eval, ax=ax1)
ax1.set_ylabel("Overall Accuracy")
sns.barplot(x='tree', y ='prec_m',data=test_eval, ax=ax2)
ax2.set_ylabel("precision of seagrass and seaweed")
plt.suptitle('trained with 4 images')
```

2.3 parameter adjustment : training size

```
# adjust training size.   tree number fixed
test_oa = []
train_size = []
moba_prec = []
higata_prec = []
water_prec = []
sns.set_style('whitegrid')

# use same test dataset for each classifier
from sklearn.model_selection import train_test_split
Feature,TEST_FEATURES, Label, TEST_LABELS =
train_test_split(features, labels, test_size=0.2)
# fix this TEST_LABELS for later accuracy check

# training size = y
train_features, test_features, train_labels, test_labels =
train_test_split(Feature, Label, train_size = y)
train_size.append(train_features.shape[0])
print('Training Features Shape:', train_features.shape)

# train
classifier = RandomForestClassifier(n_estimators = 100)
classifier.fit(train_features, train_labels)

# test
west_pred = classifier.predict(TEST_FEATURES)

# accuracy, feature_importance and confusion matrix
```

```
print('accuracy is:',accuracy_score(TEST_LA-
BELS,west_pred),'\n')
test_oa.append(accuracy_score(TEST_LABELS,west_pred))

print("confusion matrix:",'\n',confusion_matrix(TEST_LA-
BELS, west_pred, labels=['moba','higata','water']),'\n')

from sklearn.metrics import precision_score
moba_prec.append(precision_score(TEST_LABELS, west_pred,
labels=['moba'], average=None)[0])
higata_prec.append(precision_score(TEST_LABELS, west_pred,
labels=['higata'], average=None)[0])
water_prec.append(precision_score(TEST_LABELS, west_pred,
labels=['water'], average=None)[0])

case = pd.DataFrame(np.arange(1,len(train_size)+1,1))
test_eval = pd.concat([case,pd.DataFrame(train_size),pd.Da-
taFrame(test_oa),pd.DataFrame(moba_prec),pd.DataFrame(hi-
gata_prec),pd.DataFrame(water_prec)], axis=1)
test_eval.columns =
['case','size','oa','prec_m','prec_h','prec_w']
test_eval.to_csv("trainsize_eval.csv")
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10,5))
plt.subplots_adjust(wspace=0.5)
sns.barplot(x='size', y ='oa',data=test_eval, ax=ax1)
ax1.set_ylabel("Overall Accuracy")
sns.barplot(x='size', y ='prec_m',data=test_eval, ax=ax2)
ax2.set_ylabel("precision of seagrass and seaweed")
for ax in fig.axes:
    plt.sca(ax)
    plt.xticks(rotation=90)
plt.suptitle('trained with 4 images')
```

3 train classifier with best parameter setting

```
# training size = 13000
from sklearn.model_selection import train_test_split
```

```
train_features, test_features, train_labels, test_labels =
train_test_split(features, labels, train_size = 13000)

print('Training Features Shape:', train_features.shape)
print('Testing Features Shape:', test_features.shape)

# train
classifier = RandomForestClassifier(n_estimators = 100)
classifier.fit(train_features, train_labels)

# Feature Importance
fti = classifier.feature_importances_
imp = pd.concat([pd.DataFrame(feature_list),pd.Data-
Frame(fti)], axis=1)
imp.columns = ['feature','importance']
imp = imp.sort_values(by='importance')
sns.set_palette('summer_r')
sns.barplot(x='feature', y='importance',data=imp)
plt.title("feature importance when trained with 4 images")
```

4 apply classifier to whole study site for prediction

4.1 read satellite image, water depth and distance geotiff file into python

```
band_path = '/content/drive/Shared drives/Qin Xiaoyu
2017.10/west one-study-site/pred whole
study/whole_1027.tif'
wd_path = '/content/drive/Shared drives/Qin Xiaoyu
2017.10/west one-study-site/pred whole
study/whole_depth.tif'
dist_path = '/content/drive/Shared drives/Qin Xiaoyu
2017.10/west one-study-site/pred whole
study/whole_dist.tif'

nir = rasterio.open(band_path).read(4)
wd = rasterio.open(wd_path).read(1)
dist = rasterio.open(dist_path).read(1)

cols = nir.shape[1]
```

```
b = rasterio.open(band_path).read(1).ravel()
g = rasterio.open(band_path).read(2).ravel()
r = rasterio.open(band_path).read(3).ravel()
nir = nir.ravel()

band = pd.DataFrame(np.array([b,g,r,nir]).transpose())
depth = pd.DataFrame(wd.ravel().transpose())
distance = pd.DataFrame(dist.ravel().transpose())
```

4.2 standardization

```
image = pd.concat([distance,depth, band], axis=1)
image.columns =
['NEAR_DIST','depth','b1_planet','b2_planet','b3_planet','b
4_planet']
#image['NDVI'] = (image['b4_planet'] - image['b3_planet'])
/ (image['b4_planet'] + image['b3_planet'])
#image['GNDVI'] = (image['b4_planet'] - image['b2_planet'])
/ (image['b4_planet'] + image['b2_planet'])
#image['ratio'] = image['depth'] / image['NEAR_DIST']

orig_planet = rasterio.open("/content/drive/Shared
drives/Qin Xiaoyu 2017.10/west one-study-site/pred whole
study/whole_1027.tif")
bband = orig_planet.read(1).ravel()
gband = orig_planet.read(2).ravel()
rband = orig_planet.read(3).ravel()
nirband = orig_planet.read(4).ravel()

# standardization pixel values
mean1 = np.mean(bband[bband!=65535])
std1 = np.std(bband[bband!=65535])
print('mean and std of blue band is:', mean1, std1)
mean2 = np.mean(gband[gband!=65535])
std2 = np.std(gband[gband!=65535])
print('mean and std of green band is:', mean2, std2)
mean3 = np.mean(rband[rband!=65535])
std3 = np.std(rband[rband!=65535])
print('mean and std of red band is:', mean3, std3)
```

```
mean4 = np.mean(nirband[nirband!=65535])
std4 = np.std(nirband[nirband!=65535])
print('mean and std of nir band is:', mean4, std4)


# The standard score of a sample x is calculated as: z = (x
- u) / s
image['pb1_std'] = image['b1_planet'].apply(lambda x : (x -
mean1)/std1)
image['pb2_std'] = image['b2_planet'].apply(lambda x : (x -
mean2)/std2)
image['pb3_std'] = image['b3_planet'].apply(lambda x : (x -
mean3)/std3)
image['pb4_std'] = image['b4_planet'].apply(lambda x : (x -
mean4)/std4)
```

4.3 mask land pixels

```
land_idx = image[image['depth']==9999].index
# image2: has exclude land area, hence only should contain
water and moba.
image2 = image.drop(land_idx)
```

4.4 apply classifier to predict

```
image2 = image2[feature_list]
ori_idx = image2.index
image2_pred = classifier.predict(image2)
```

4.5 visualization

```
image2_pred_df = pd.DataFrame(image2_pred, columns=['pre-
dicted'])
image2_pred_df = image2_pred_df.set_index(ori_idx)

merged = image.merge(right=image2_pred_df, left_index=True,
right_index=True, how='left')

land_condition = (merged['predicted']!='water') &
(merged['predicted']!='moba') & (merged['predicted']!='hi-
gata')
merged.at[land_idx,'predicted'] = 'land'
```

```
pred_image_3class = merged['predicted']
pred_image_3class = np.array(pred_image_3class)

convert = np.reshape(pred_image_3class, (-1, cols))

a = pd.DataFrame(convert).replace(to_re-
place=['water','moba','higata','land'], value=[3, 2, 1, 0])
plt.figure(figsize = (7,7))
sns.heatmap(a, cmap='GnBu', cbar=False)
plt.axis('off')
plt.title('Planet, 4-image. 100-tree. 13000. pred on Oct
27')
plt.savefig("Planet, 4-image. 100-tree. 13000. pred on Oct
27.png", dpi=600)
```

5 evaluation of prediction

```
# import ground truth data for test area
true_h = rasterio.open("/content/drive/Shared drives/Qin
Xiaoyu 2017.10/west one-study-site/pred whole
study/h_whole_true.tif").read(1)
true_m = rasterio.open("/content/drive/Shared drives/Qin
Xiaoyu 2017.10/west one-study-site/pred whole
study/m_whole_true.tif").read(1)

trueh_rav = true_h.ravel()
truem_rav = true_m.ravel()

merged['ish'] = trueh_rav
merged['ism'] = truem_rav

merged_3class = merged[merged['predicted']!='land']
is_higata = merged_3class['ish'] != 9999
is_moba = merged_3class['ism'] != 9999
merged_3class.at[is_higata,'class'] = 'higata'
merged_3class.at[is_moba,'class'] = 'moba'
```

```
merged_3class['class'] = merged_3class['class'].re-
place(np.nan, 'water', regex=True)

target_names = ['moba', 'higata', 'water']

pred_cm = confusion_matrix(merged_3class['class'],
merged_3class['predicted'])
print(pred_cm)
sns.heatmap(pred_cm, cmap='Blues')
plt.show()

from sklearn.metrics import cohen_kappa_score
print(cohen_kappa_score(merged_3class['class'],
merged_3class['predicted']))
pred_kappa.append(cohen_kappa_score(merged_3class['class'],
merged_3class['predicted']))

from sklearn.metrics import classification_report
y_true = merged_3class['class']
y_pred = merged_3class['predicted']
report = classification_report(y_true, y_pred, out-
put_dict=True)
df1027 = pd.DataFrame(report).transpose()
```

Acknowledgement

This thesis is part of the Education and Research Program of PlanetScope. Without access to the fantastic satellite imagery product, this study could not have been conducted.

I would first like to thank my thesis advisor, professor Jun Sasaki. Sasaki sensei was always keen to provide his honest and insightful advice about my research or writing. He taught a lot about how to become a qualified researcher, how to do work that is valuable to the society, how to manage to think in a more critically way, and, most of all, to have passion and guts to pursue more challenging work.

I would also like to thank my co-supervisor, professor Ikuho Yamada, who was involved in the review session for this research project. Without her wise suggestion on how to improve the content of the paper itself and how to make a more straightforward and understandable presentation, the output could not have been successfully conducted.

Besides my supervisors, my sincere thanks also go to my colleagues: Masanori Endo, Fei Liu, Jundong Chen, Suxing Lv, Ayumi Furuhata, Mirei Fujisaki, Win Sithu Maung for their insightful comments and encouragement. Talking with them about the research really was a fascinating experience, and thanks to this, it had widened my research from various perspectives.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Xiaoyu Qin