DISCOV: A Neural Network Classifier for Intertidal Vegetation Drone Mapping

2024-11-21

Summary

Coastal intertidal seagrass meadows are critical ecosystems providing essential services such as carbon sequestration, erosion control, and habitats for diverse marine species [@sousa2019blue; @unsworth2022]. Despite their importance, these ecosystems face increasing threats from human activities such as eutrophication, and habitat fragmentation, which diminish their ecological functions [@chefaoui2018dramatic; @nguyen2021; @wang2022]. Monitoring these habitats has been challenging due to their dynamic nature and the complexity of distinguishing vegetation types with similar pigment compositions [@Douay2022; @ralph2002]. This study addresses these challenges by developing a deep learning model capable of accurately classifying macrophytes, based on drone multispectral images. The model was trained on high-resolution drone-based reflectance data collected across diverse intertidal habitats in France and Portugal, encompassing monospecific and mixed vegetation meadows. It successfully discriminates between macrophyte types, leveraging subtle spectral differences, even in challenging cases where pigment similarity could hinder classification, for example between seagrasses (Magnoliopsida) and green macroalgae (Chlorophyceae) [@Davies2023]. This approach highlights the potential for advanced modeling techniques to improve vegetation mapping accuracy, offering insights into intertidal ecosystem dynamics and supporting more effective conservation strategies.

Statement of needs

Accurately mapping and classifying intertidal vegetation, such as seagrasses and green macroalgae, is vital for the sustainable management of coastal ecosystems. These habitats provide critical services, including carbon sequestration, erosion control, and biodiversity support, but face escalating threats from anthropogenic pressures, climate change, and eutrophication. Monitoring these ecosystems requires precise, reliable, and scalable tools to assess habitat distribution, health, and changes over time.

Traditional monitoring methods, such as field surveys and satellite remote sensing, face significant limitations. Field surveys are labor-intensive, restricted by tidal accessibility, and often lack spatial coverage. Satellite data, while providing broader coverage, typically struggles with inadequate spatial and spectral resolution, making it difficult to differentiate between vegetation types like seagrasses and green macroalgae, which share similar pigment compositions and spectral characteristics. This gap necessitates innovative solutions to deliver high-resolution and accurate mapping.

The **DISCOV (Drone Intertidal Substrate Classification Of Vegetation)** model offers a groundbreaking solution to these challenges. By leveraging drone-based high-resolution multispectral imagery and a neural network classifier, DISCOV enables the precise discrimination of vegetation types, even in mixed and heterogeneous habitats. This model delivers several capabilities:

- **High Classification Accuracy**: DISCOV achieves a classification accuracy of 94%, with 97% accuracy in distinguishing seagrasses from green macroalgae, addressing a key challenge in remote sensing applications.
- **Disctinction of 5 vegetation classes**: DISCOV is not only able to distinguish green algae from seagrasses, but can also recognize the most common vegetation classes found in intertidal areas along the European Atlantic coastline: Phaeophyceae (Brown macroalgae), Rhodophyceae (Red macroalgae) and Bacillariophyceae (diatoms).

Applications of DISCOV span a range of critical environmental management and research areas:

- **Early Detection of Green Tides**: Monitoring and identifying the initial stages of algal blooms to mitigate their ecological and economic impacts.

- **Seagrass Health Assessment**: Evaluating resilience to eutrophication, climate stressors, and other anthropogenic pressures.

- **Policy Support**: Providing actionable data to meet conservation goals outlined in international frameworks like the EU Water Framework Directive of the Marine Strategy Framework Directive.

- **Biodiversity Conservation**: Mapping habitats to follow up on restoration and environmental protection efforts.

DISCOV is addressing the need for advanced tools to combat biodiversity loss and environmental degradation. By integrating drone-based insights with larger-scale remote sensing efforts, DISCOV offers a scalable, efficient, and accurate framework for understanding and managing intertidal vegetation dynamics in the face of accelerating global change.

Material & Methods

DISCOV development employed drone-based high-resolution multispectral imagery to classify intertidal vegetation across nine sites in France and Portugal. Two flight altitudes (12 m and 120 m) were used, providing spatial resolutions of 8 mm and 80 mm, respectively. A DJI Matrice 200 drone equipped with a Micasense RedEdge Dual MX sensor (10 spectral bands: 444–840 nm) was used for image acquisition. Ground control points were established with differential GPS for georeferencing, and reflectance was calibrated using a standard panel.

Low-altitude flights were utilized to train the DISCOV model, a neural network classifier developed using the fastai workflow with two hidden layers and 26,054 trainable parameters. The model was trained on more than 418,000 pixels from the 8 mm resolution imagery, categorized into seven classes: seagrasses, green macroalgae, red macroalgae, brown macroalgae, benthic diatoms, sediment, and water. Training incorporated spectral bands, their standardized values, and the Normalized Difference Vegetation Index (NDVI).

Validation was performed using both low- and high-altitude flights, with 536,000 pixels collected from *in situ* field measurements and photo-interpretations. Metrics such as global accuracy, sensitivity, specificity, F1 score, and Kappa coefficient were calculated to evaluate model performance.

How to install DISCOV ?

DISCOV was developed in Python programming language. The current repositories of DISCOV include Python code, used to train the model and make predictions on images, as well as R code used to update the training dataset with your own data and to plot the results of the predictions.

Windows

The first step is to install all the necessary software, if you have not done so already. You will need to have R and Conda installed on your computer. For Conda, we recommend installing it directly through Conda-forge using the Windows installer, which can be downloaded here. R can be downloaded from this link. We recommend using RStudio, an integrated development environment specifically designed for R. Alternatively to RStudio, you can use software like VS code which allows you to edit and run code written in both R and Python within the same working environment. Once Conda is installed, open the terminal
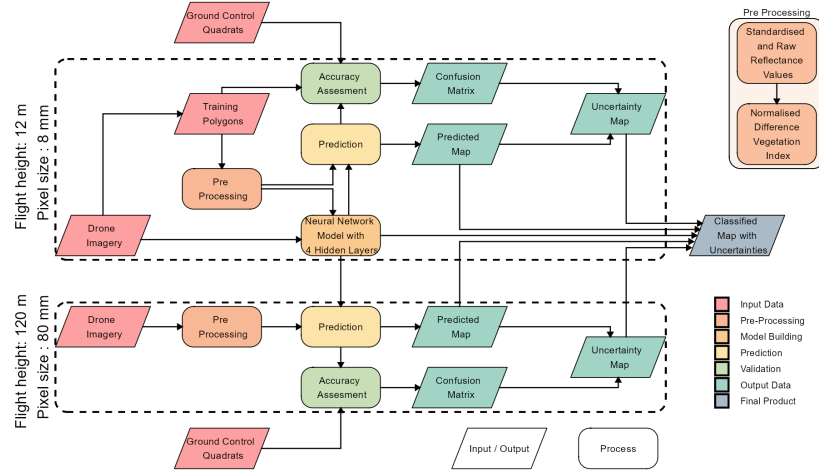
Figure 1: Schematic representation of the workflow. Parallelograms represent input or output data, and rectangles represent Python processing algorithms. The overall workflow of this study is divided into two distinct parts based on the spatial resolution of the drone flights: high-resolution flights (pixel size: 8 mm) were used for training and prediction of the Neural Network model, whereas lower-resolution flights (pixel size: 80 mm) were solely employed for prediction and validation purposes.
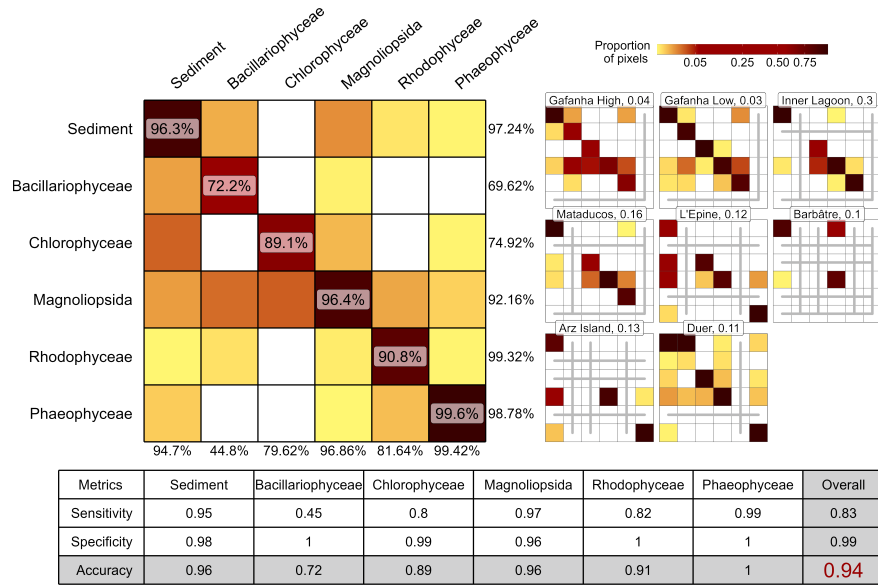
Figure 2: A global confusion matrix on the left is derived from validation data across each flight, while a mosaic of confusion matrices from individual flights is presented on the right. The labels inside the matrices indicate the balanced accuracy for each class. The labels at the bottom of the global matrix indicate the User's accuracy for each class, and those on the right indicate the Producer's Accuracy. The values adjacent to the names of each site represent the proportion of total pixels from that site contributing to the overall matrix. Grey lines within the mosaic indicate the absence of validation data for the class at that site. The table at the bottom summarizes the Sensitivity, Specificity, and Accuracy for each class and for the overall model.

(Press windows + R and enter "cmd" to open the terminal). Typing `where conda` should give a reply. If at this step you have an error, check your conda installation. Now you can clone the repositories using this link. Save and extract it to a safe location. Once that is done, you can open the terminal in the folder by pressing the Ctrl key on your keyboard and right-clicking in the folder, then select 'Open terminal'. Entering `conda env create -f environment.yml` will install the correct version of Python and all the dependencies needed for DISCOV to run. If you encounter an error during the installation of pip dependencies, make sure you have enabled long path support in Windows.

Once that is done, you should be ready to work with the model. You can activate the environment by typing `conda activate NN_env` in the terminal, or by selecting the NN_env kernel in Visual Studio Code.

MAC OS

The first step is to install all the necessary software, if you have not done so already. You will need to have R and Conda installed on your mac. For Conda, we recommend installing it directly through Conda-forge using the MacOSX installer, which can be downloaded here. To run this script, open the terminal by pressing Command + Space and navigate to the Downloads folder using the `ls` and `cd` commands. Then, run `sh Miniforge3-MacOSX-x86_64.sh` and follow the instructions. Once it is complete, press 'yes' or execute `conda init` and close the Terminal. R can be downloaded from this link. We recommend using RStudio, an integrated development environment specifically designed for R. Alternatively to RStudio, you can use software like VS code which allows you to edit and run code written in both R and Python within the same working environment. Once Conda is installed, open the terminal by pressing Command + Space and typing 'terminal'. In the terminal, type `where conda` to check if it is installed correctly. If you receive a 'conda not found' message at this step, verify your Conda installation. Now you can clone the repositories using this link. Save and extract it to a safe location. Once that is done, you can open the terminal and navigate to the DISCOV-MicaSense folder using the `ls` and `cd` commands. Entering `conda env create -f environment_MACOSX.yml` will install the correct version of Python and all the dependencies needed for DISCOV to run.

Once that's done, you should be ready to work with the model. You can activate the environment by typing `conda activate NN_env` in the terminal, or by selecting the NN_env kernel in Visual Studio Code.

How to use DISCOV ?

The code of DISCOV is located in the "Scripts" folder.

Training dataset building

The original training dataset for DISCOV is located in Data/Training/DISCOV_BiCOME_Training.txt. This file contains a table with 22 columns: 21 predictors used by the model and the class assigned to each pixel, totaling over 450,000 pixels. The training dataset can be updated using the R script named `Updating_Training_Data.qmd`. This script requires two inputs: a new Micasense image, pre-processed in reflectance and including the 10 spectral bands, from which additional training pixels will be extracted, and a polygon shapefile containing a column labeled "True_Class" that specifies the class information for each polygon. The output of the script is a text file that combines the original DISCOV training dataset with the newly generated training data.

Train a New version of DISCOV

The script `Model_Training.ipynb` contains Python code designed to train and create a model from a given training dataset. It takes as input a text file containing the training dataset and outputs a pickle file (`.pkl`) with the newly trained model. Additionally, the script includes functionality to perform a variable importance analysis on the model.

Predict on a new image

The scripts `Prediction.ipynb` and `Batch_prediction.ipynb` are used to make predictions on new Micasense images. `Prediction.ipynb` is designed for a single image, while `Batch_prediction.ipynb` is made for processing multiple images simultaneously. The image must be placed in the Data/img directory, and its name should be specified in the first section of each script. Additionally, the name of the model to be used and the desired name of the output file must also be defined in the first section of the code. These scripts produce two raster outputs for each image input: - Prediction Raster: Assigns a class value to each pixel in the image. - Probability Raster: Represents the model's confidence level for each pixel.

The version 1.0 of DISCOV gives has output a TIFF file with values ranging between 1 and 10:

- **1 - Microphytobenthos**: Film of unicellular microalgae and/or Cyanobacteria colonizing superficial sediments at low tide. These biofilms can cover up to several square kilometers. The primary class of microalgae forming these biofilms is Bacillariophyceae, commonly known as diatoms.

- **2 - Chlorophyceae**: Green algae from the genus *Ulva sp* stranded on the sediment. The model has primarily been trained on *Ulva lactuca*, *Ulva armoricana*, and *Ulva intestinalis* . Therefore, We are uncertain how the model will behave when encountering other types of Chlorophyceae outside of the Ulvophyceae class, such as *Caulerpa sp.*).

7

- **3 - Magnoliopsida**: Marine angiosperm of the genus *Zostera sp.*. The model has been trained exclusively on pixels from *Zostera noltei.*

- **4 - Phaeophyceae**: Brown macrolalgae, stranded on the sediment or oftenly attached to rocks. Mainly trained from pixels of the genus *Fucus sp.*

- **5 - Rhodophyceae**: Red Algae. This class was trained with the fewest pixels in DISCOV V1.0, using only pixels from *Gracilaria sp.* (likely *Gracilaria vermiculophylla*) observed in the Ria de Aveiro coastal lagoon, Portugal.

- **6 - Bare Sediment**: This class is primarily trained on bare mud but also performs well on sand. It can be mistaken for Microphytobenthos because sometimes the bare mud contains a small amount of chlorophyll-a, which absorbs light around 668 nm.

- **7 - Sun Glint**: Depending on the solar angle at the time of the flight, some pixels receive specular reflections directly from the sun, leading to an overestimation of the pixel's total reflectance and distorting the spectral shape. This 'sun glint' class has been trained to prevent pixels affected by glint from being incorrectly classified as a type of vegetation. When there is residual water on the surface of the sediment, the probability of encountering glinted pixels increases.

- **8 - Water**: When water is shallow and vegetation is present at the bottom, the spectral signature of the vegetation changes based on the thickness of the overlying water layer, especially in the infrared spectrum. This can lead to incorrect pixel classification. This class was primarily established to avoid such scenarios by ensuring that very shallow waters are correctly classified as water. It is also effective in handling optically deep waters.

- **9 - NA**

- **10 - NA**

Acknowledgments

References