Rugulopterix okamurae

Approaches to Spatio-Temporal Detection

By: *Gayathri Girish Nair (girishng@tcd.ie), Hafssa Najouh (hafssa.najouh@ucdconnect.ie), Rogier Putker (rogier.putker@ucdconnect.ie), Aditi Paretkar (aditi.paretkar@ucdconnect.ie), Yongyao Liang (yongyao.liang.2025@mumail.ie), Tingting Kang (tingting.kang@ucdconnect.ie), Muhammad Bin Arif (muhammad.b.arif@ucdconnect.ie), and Aidan Magee (aidan.magee@mu.ie).* ***For:*** *Zinto Labs*

Abstract

Rugulopteryx okamurae, a species of brown macroalgae native to Japan and Korea, has become highly invasive in the Mediterranean Sea and Atlantic ocean. It remains very difficult to mitigate once established (Laamraoui et al., 2024). Thus, faster or more reliable detection and early detection is paramount to limiting serious degradation of native ecosystems.

This work involved a surface-level study of species characteristics / preferences, existing datasets, few existing approaches to seaweed detection, and solutions to similar challenges in other fields over a course of 4 days, culminating in the *Regulopteryx okamurae* Spatio-Temporal (ROST) Early Detection Model, a proposed (theoretical) Graph Neural Network (GNN) based solution to the underlying problem of **spatio-temporal propagation prediction**. A quick analysis of practicalities around development of the ROST model and gathering of data requirements therein is also presented here.

Furthermore, simple approaches to presence detection based on existing standard computer vision methods like the Convolutional Neural Network (CNN) for object detection applied to remote sensing data from the Copernicus Sentinel satellites fetched via Google Earth Engine were explored in addition to working with basic ML and Data Science (DS) algorithms to identify patterns in limited occurrence data fetched from GBIF (Global Biodiversity Information Facility) which resulted in a map of geolocations ranked according to habitat suitability for *R. okamurae*.

Please find all code and literature reviewed in relation to this work on GitHub. But beyond results at face value, the primary aim of this report is to leave the reader with promising directions for further investigation (highlighted in pink) that can lead to a practical solution to early detection of highly invasive species like *R. okamurae* and / or may be used to inform mitigation efforts.

Code

Please find code associated with this work on GitHub.

Data

The following table summarizes data used with code presented later in this document. This data was used instead of others mostly due to practical convenience w.r.t accessibility, storage and download speed.

Data	**Description**	**Source**
Occurrence data.	*R. okamurae* presence and absence data. After removal of duplicates and points with missing "year" information, the dataset contained 489 instances of *R. okamurae* presence and 32 instances of its absence.	Global Biodiversity Information Facility (GBIF). (GBIF.org (3 November 2025) GBIF Occurrence Download https://doi.org/10.15468/dl.sprd2u)
Remote sensing data.	Bands **Oa08_radiance**, **Oa17_radiance**, and **Oa21_radiance** where band 8 can help with detecting vegetation while bands 17 and 21 can aid in detecting "not" vegetation.	Google Earth Engine (GEE) database "Sentinel-3 OLCI EFR: Ocean and Land Color Instrument Earth Observation Full Resolution".
Oceanographic Data	Sea surface temperature (°C), salinity (PSU), Chlorophyll-a concentration (mg/m³), Dissolved oxygen (mL/L), Primary productivity (gC/m²/day), and pH.	Bio-Oracle Database.

Initial exploration involved browsing of other sources in addition to ones in the table above, such as Sentinel-2 and datasets mentioned in explored literature (as in the literature folder on GitHub).

Resolution and clarity of visibility from space is a key challenge when it comes to seaweed classification using satellite data. Since radar is less distorted by cloud cover and fog, we considered using data from Synthetic Aperture Radar (SAR) satellites, but it could not be fetched from GEE in time. However, this remains an interesting avenue for future experimentation.

Habitat Suitability Analysis

This study aimed to assess the environmental suitability and potential global distribution of the invasive brown algae *Rugulopteryx okamurae* using a combination of occurrence records and environmental variables extracted from the **Bio-Oracle** dataset.

The dataset was first cleaned, filtered, and enriched with key environmental parameters such as sea surface temperature, salinity, chlorophyll concentration, pH, and dissolved oxygen. Correlation analyses were conducted to detect and remove redundant predictors, ensuring that only the most relevant variables were retained for modeling.

A **Logistic Regression model** was then applied to predict the probability of habitat suitability (ranging from 0 to 1) for *Rugulopteryx okamurae* across global marine environments. The model effectively discriminated between suitable and unsuitable regions, highlighting high suitability values in areas corresponding to both **the native range (Japan and the western Pacific)** and **the invaded regions (European Atlantic coasts — France, Spain, Portugal)**. Additional zones with moderate suitability, such as **South America** and **southern Australia**, were identified as potentially vulnerable to future colonization.

The resulting maps provide valuable insights into the **ecological preferences** of the species, indicating a strong association with **temperate waters**, **moderate salinity**, and **high productivity zones**. These conditions appear to play a critical role in defining its potential spread.

Overall, the integration of geospatial data, environmental predictors, and statistical modeling successfully demonstrated how **species distribution modeling** can be used as a **predictive tool for early detection and risk assessment** of marine invasive species. This approach contributes to improving **monitoring strategies** and supports **preventive management actions** to mitigate ecological and economic impacts associated with *Rugulopteryx okamurae*.

Data Preparation and Processing of Environmental and Biological Variable*s*

Two main data sources were used as follows.

- **Biological Data:** Occurrence points of *Rugulopteryx okamurae* obtained from biodiversity databases such as the Global Biodiversity Information Facility (GBIF).
- **Environmental Data:** Oceanographic variables extracted from the Bio-Oracle database, including:

- Sea surface temperature (°C).
- Salinity (PSU).
- Chlorophyll-a concentration (mg/m³).
- Dissolved oxygen (mL/L).
- Primary productivity (gC/m²/day).
- pH.

These variables were selected because of their ecological importance in influencing the growth and spatial distribution of marine algae.

Correlation Analysis Between Environmental Variables and Species Presence

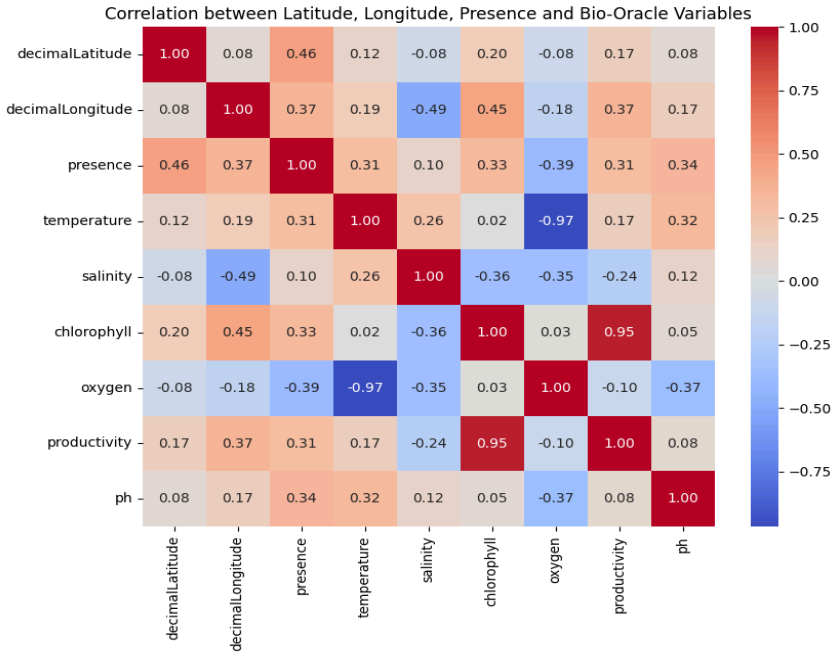

Figure 1. Correlation between Latitude, Longitude, Presence and Bio-Oracle Variables

This matrix shows how the species' presence relates to spatial position (latitude, longitude) and environmental conditions.

- **Presence Correlations:**
  - Moderate positive correlations with latitude (0.46) and longitude (0.37) indicate a spatial pattern in the species' distribution.
  - Positive correlations with temperature (0.31), chlorophyll (0.33), and pH (0.34) suggest that Rugulopteryx okamurae tends to occur in warmer, nutrient-rich, and slightly basic waters.

- Negative correlation with oxygen (-0.39) indicates a preference for areas with lower oxygen levels, typical of warm and biologically active waters.

- **Environmental Interrelations:**
  - Temperature and oxygen (-0.97) show a strong negative correlation — as expected, warmer waters hold less dissolved oxygen.
  - Chlorophyll and productivity (0.95) are highly correlated, reflecting that productive areas coincide with high phytoplankton activity.
  - Longitude and salinity (-0.49) suggest possible east–west salinity gradients.

Environmental Suitability Mapping for *Rugulopteryx okamurae*

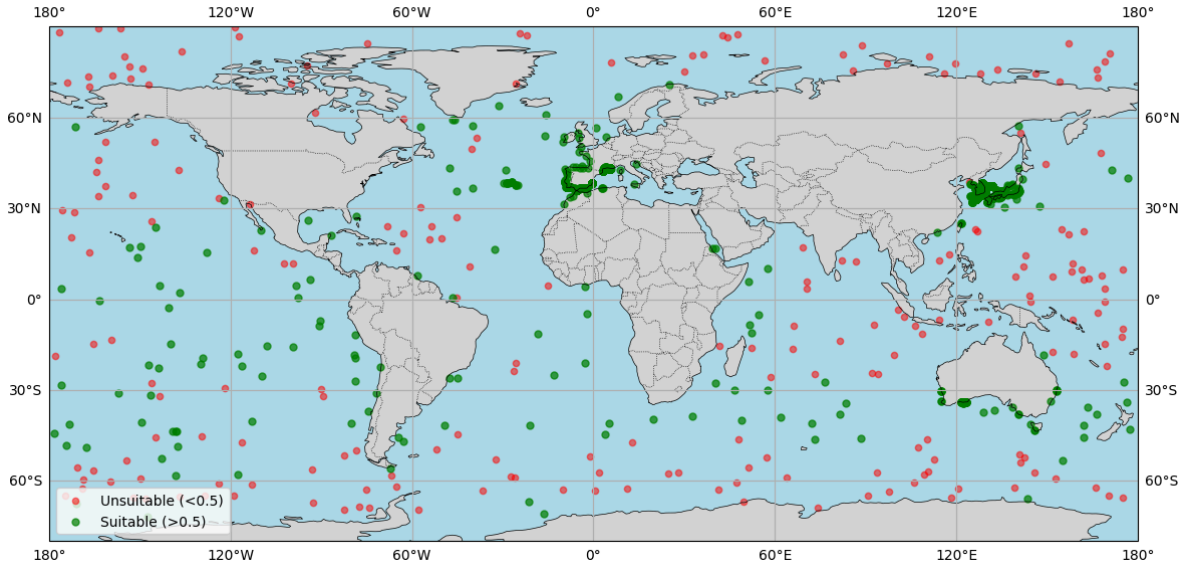

Figure 2. Environmental Suitability Mapping for *Rugulopteryx okamurae*

This analysis provides a spatial prediction of habitat suitability based on the environmental variables extracted from the Bio-Oracle dataset,highlights both **current and potential future habitats** for *Rugulopteryx okamurae*.The goal of this step is to identify and visualize the regions of the world where environmental conditions are potentially suitable for the establishment of *Rugulopteryx okamurae*.

Regions identified as suitable should be considered **priority monitoring zones** for early detection and management efforts.

This spatial modeling approach supports preventive strategies by predicting potential invasion hotspots under current environmental conditions.

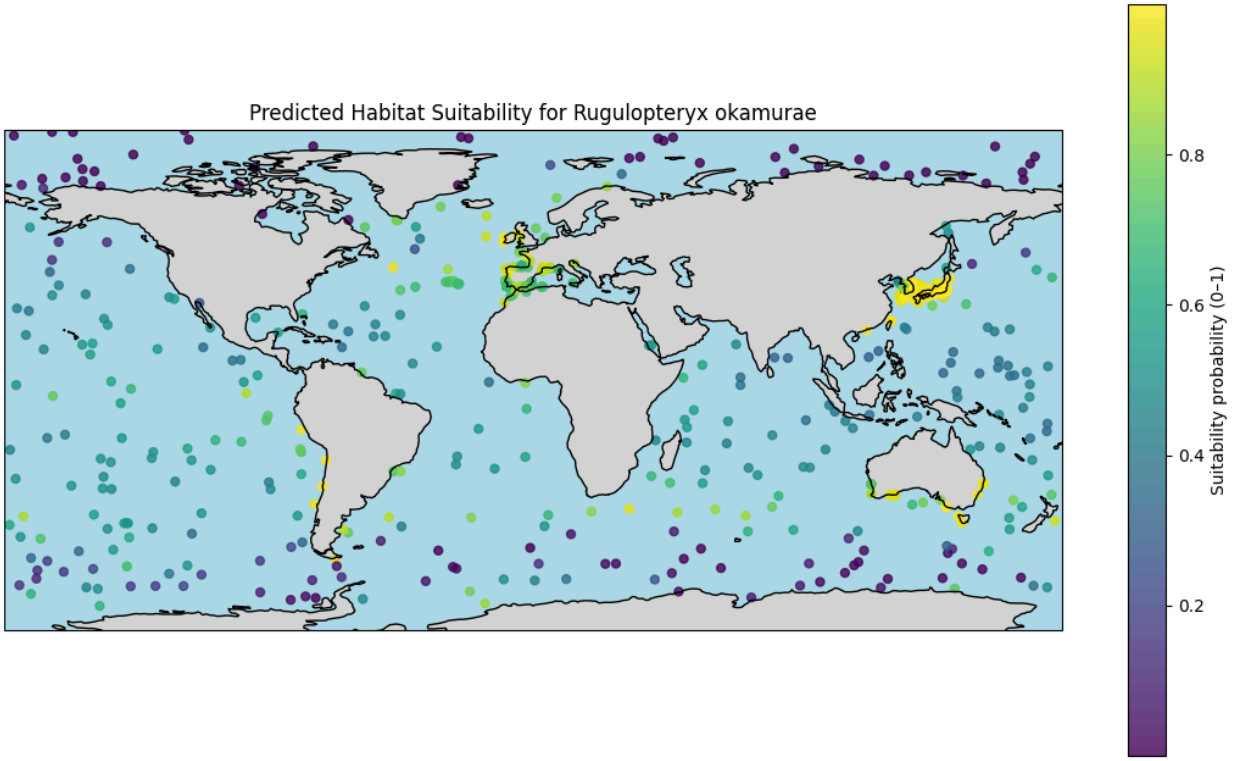

Figure 3. Habitat suitability mapping for *Rugulopteryx okamurae.*

The map above illustrates the spatial distribution of predicted habitat suitability across the globe.

Key Observations:

- **High suitability** (yellow zones, probability > 0.6) is found along the **European Atlantic coasts** (mainly France, Spain, and Portugal), consistent with the current invaded range.
- **Native regions** such as **Japan** and **the western Pacific** also exhibit high suitability values.
- Moderate suitability areas appear in **South America**, **southern Australia**, and parts of **northern Africa**, indicating potential future spread zones.
- Open ocean areas and high-latitude zones display low suitability values (purple), reflecting unfavorable environmental conditions.

Ecological interpretation:

The model suggests that *Rugulopteryx okamurae* prefers:

- Warm, temperate waters.
- Moderate to high primary productivity.

- Slightly basic pH and stable salinity.

These findings align with the environmental profile observed in both the native and invaded habitats of the species.

Please find code associated with habitat suitability analysis at `habitat_suitability_analysis/main.ipynb` on GitHub.

Detection - Presence Classification - CNN

A Convolutional Neural Network (CNN) is a type of feedforward Neural Network (NN) that can learn to extract features from images or image-like 3D input with a shape akin to [Channels x Height x Width] whilst considering spatial context. This is achieved by allowing a Neural Network to learn the numbers in several 2D kernels. Here, kernels refer to smaller matrices that can be slid over the image (which is also a matrix, just a bigger one) deterministically to perform a convolution operation after each stride.

A convolution operation in this context, involves summing up element wise products between each kernel matrix cell value and the associated image patch matrix value to obtain an aggregate number such that certain pixels in the original image may have contributed more to this value than others (Sanderson & 3Blue1Brown, 2022). When repeated over the entire image, this operation can result in certain image features being highlighted over others. Depending on kernel configuration (magnitude of numbers in each kernel cell and what they all add up to), convolution over images can lead to extraction of meaningful features like outlines foredge detection, a smoother version of the image for blurring, etc (Sanderson & 3Blue1Brown, 2020). In computer vision related ML, the aim is for the model to be able to "learn" features from images that are most useful in predicting the given target variable. To this end, CNNs are designed to allow for the configuration of several kernels to be learned such that together after training, they can pick up on features in input images that can reliably inform target variable predictions.

CNNs are standard when it comes to object detection and feature extraction from images. Thus, using a CNN to detect presence or absence of *R. okamurae* from satellite images is a sound approach.

Dataset Preparation

Image classification requires a labelled dataset of satellite image with associated labels of whether the algae is present = 1 or absent = 0. This was achieved as follows.

1. Download requests were made for data from GBIF using code at `/data/extraction/occurrences_gbif/get_from_gbif.py` on GitHub. The data so obtained is available at `data/db/occurrence/raw/gbif_0013072-251025141854904.csv` on GitHub. The `get_from_gbif.py` script required knowledge of the GBIF taxon key (unique identifier in the GBIF DB) associated with *R. okamurae*. This was found by uploading the file at `/data/extraction/occurrences_gbif/needs_gbif_key.csv` to the online GBIF species lookup tool which returned file at `/data/extraction/occurrences_gbif/species_gbif.csv` containing this information in column "key".
2. Occurrence status "PRESENT" and"ABSENT" were separated from each other, filtered to remove duplicates and retain only useful columns (longitude, latitude, year) as well as rows with no missing data. This was done using code at `data/extraction/occurrences_gbif/extract_present_absent.ipynb` and resulted in "present.csv" and "absent.csv" at `/data/db/occurrence/processed` on GitHub.
3. Mean (μ) and standard deviation (σ) of 3 bands (Oa08_radiance, Oa17_radiance, and Oa21_radiance) from the *Sentinel-3 OLCI EFR: Ocean and Land Color Instrument Earth Observation Full Resolution* dataset was fetched for 3 global Regions Of Interest (ROI) based on observed clustering of coordinates on a map (see Figure 4). Useful columns from GBIF occurrence data were *decimalLongitude*, *decimalLatitude*, *year*, and *occurenceStatus* (PRESENT / ABSENT). Aggregation (μ, σ) was thus performed over the year associated with each occurrence instance due to the lack of availability of more precise timestamps. Download requests were made for .tif files with 6 channels (2 aggregate values of 3 bands) corresponding to years present in both the GEE dataset and occurrence table using the GEE Online Code Editor code. Please find this code at `/data/extraction/gee/code_editor.txt` on GitHub.

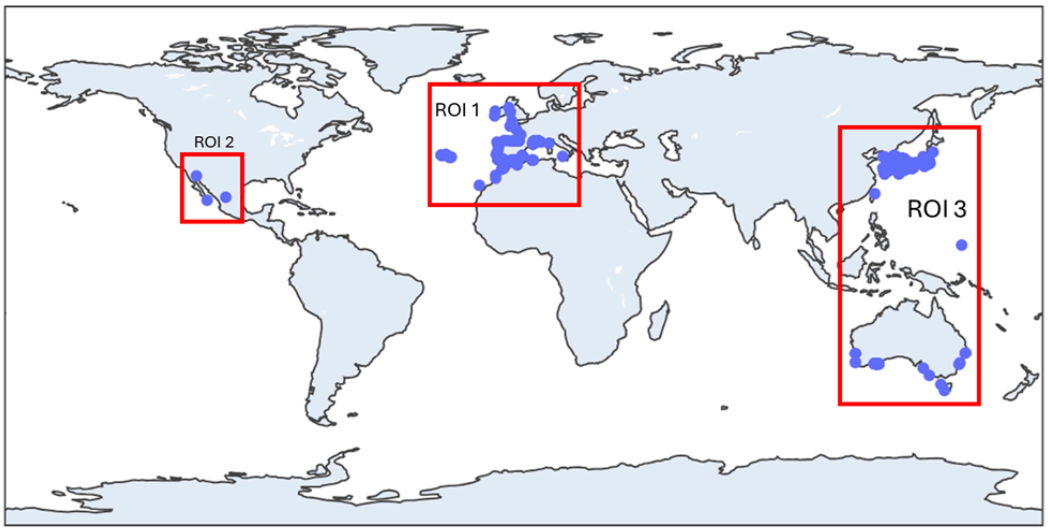

Figure 4. Visual representation of global occurrences (both presence & absence) from GBIF plotted on a map, divided into 3 regions of interest.

4. Several cropped 64 by 64 pixel patches around each occurrence (lon, lat) point needed to be extracted from downloaded .tif images for both presence and absence data. This was done using python code at `/data/extraction/gee/get_cropped_images.ipynb` on GitHub. The resulting image patches are available at `/data/db/trn_val_tst/raw` in folders `patches_present` and `patches_absent` on GitHub.
5. Next, images from both `patches_present` and `patches_absent were extracted and` placed in a single folder at `/data/db/trn_val_tst/processed` such that the target label (present = 1 / absent = 0) was made part of the file name using code at `/data/extraction/gee/view_combine_patches.R`.

RoCNN

The CNN architecture designed for *R. okamurae* presence detection referred to here, as **RoCNN** is as presented in table 1 below. It is inspired from (similar no. of layers) AlexNet (8 layer CNN that won the ImageNet Large Scale Visual Recognition Challenge in 2012) adapted to a binary classification task (sigmoid activation instead of softmax in final layer) and to be functional with input images of size 64 by 64 pixels with 6 channels.

Layer		Feature Map	Size	Padding	Kernel Size	Stride	Activation
Input	Image	1	6 X 64 X 64	-	-	-	-
CV1	Conv	32	32 X 31 X 31	0	4 x 4	2	relu
MP1	Max Pool	32	32 X 15 X 15	0	3 X 3	2	-
CV2	Conv	64	64 X 15 X 15	1	3 X 3	1	relu
CV3	Conv	64	64 X 15 X 15	1	3 X 3	1	relu
CV4	Conv	64	64 X 15 X 15	1	3 X 3	1	relu
MP2	Max Pool	64	64 X 7 X 7	0	3 X 3	2	-
Flatten	Flatten	64	3136	-	-	-	-
FC1	Dense	-	3136	-	-	-	relu
FC2	Dense	-	512	-	-	-	relu
FC2	Dense	-	64	-	-	-	relu
Output	Dense	-	1	-	-	-	sigmoid

Table 1. RoCNN architecture.

The model was trained for 50 epochs using an Adam optimizer (learning rate = 0.001) and Binary Cross Entropy (BCE) loss function. It was trained using 80% of the dataset (416 instances), validated on another 10% (52 instances) and tested on the remaining 10% (53 instances) in batches of 32 instances.

Image pre-processing, as managed alongside data fetching and splitting by the `DataDirector` object, involved the following.

1. Min-max normalization so that all values were in the [0, 1] range. This is so that uneven scales of values in each image channel does not mislead the model into assigning greater importance to channels with larger numbers simply because of greater magnitude of numbers instead of true relevance w.r.t prediction. Min and max values per samples were computed after iterating through all data points prior to training during initialization of the `DataDirector` object.
2. Padding all images so that they are all of the same size. This had to be done because given the implemented downloaded .tif images to image patches cropping logic, a small no. of patches from the edges of ROIs ended up smaller than the expected 64 by 64 pixels by few pixels.
3. Missing values, (very few if any) were filled with 0s.

Figure 5 below, presents training and validation curves showing BCE loss and accuracy over time.

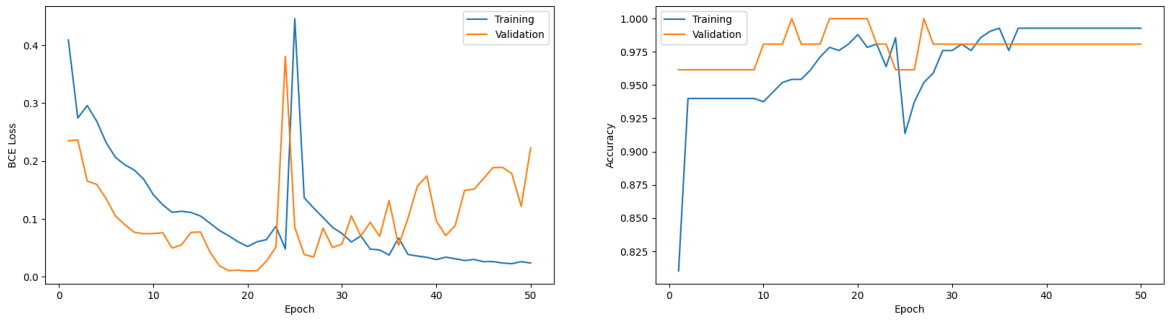

Figure 5. BCE loss and accuracy training and validation learning curves corresponding to training of the RoCNN model.

The decreasing loss and improving accuracy serves as good evidence for learning. The spikes between epochs 20 and 30 are likely due to oscillation around a local minimum in the loss landscape which the model escapes as evident from slight overfitting (training loss < validation loss) observed after around epoch 33.

Performance metrics computed other than BCE loss used for training are as follows.

- True Positive (TP) count.
- True Negative (TN) count.
- False Positive (FP) count.
- False Negative (FN) count.
- Accuracy = (TP + TN) / (TP + TN + FP + FN) = Of all predictions, how many are correct?.
- Precision = TP / (TP + FP = of all positive predictions, how many were actually positive?
- Recall = TP / (TP + FN) = of all positive data instances, how many were predicted to be positive?
- F1 Score = 2 * ((precision * recall) / (precision + recall)) = balance between precision and recall such that higher values are better.

Other metrics indicative of classification performance were computed because accuracy alone can be misleading when classes are imbalanced as is the case here (489 instances available for "present" and only 32 instances available for "absent") since accuracy can be high even if all the model does is predict the majority class every time.

Such a model that always predicts present = 1 irrespective of input was implemented to obtain baseline performance values against which to compare test results as in Figure 6 below.

Trained RoCNN BCE loss is notably lower than that of the baseline model which shows that true learning has indeed occurred as further confirmed by all the classification performance related metrics being higher in case of RoCNN as well.

```
BASELINE (ALL ONES) ===
bce loss: 9.4340
precision: 0.9057
recall: 1.0000
f1: 0.9505
accuracy: 0.9057
counts: TP=48 TN=0
        FP=5 FN=0
```

```
TEST SUMMARY ===
bce loss: 0.0101
precision: 1.0000
recall: 1.0000
f1: 1.0000
accuracy: 1.0000
counts: TP=50 TN=3
        FP=0 FN=0
```

Figure 6. Comparison of baseline v/s RoCNN performance after the latter trained for 50 epochs.

That said, the difference in more reliable metrics like F1 score is minimal (difference = 0.05) and not nearly as much as the difference in BCE loss (difference = 9.4239). This is expected to be due to high imbalance of instances among classes and limited no. of data points that the model had to work with.

Thus, creation of large, good quality labeled datasets for *R. okamurae* detection is paramount to being able to develop viable Deep Learning (DL) solutions.

All code related to RoCNN is available at `/RoCNN` on GitHub.

Towards Early Detection - Infestation Prediction - UNet CNN

Predicting how an infestation may grow and develop in future timesteps given past timesteps of a swatch of the sea, as illustrated in Figure 7, is one step more advanced than simple presence detection.

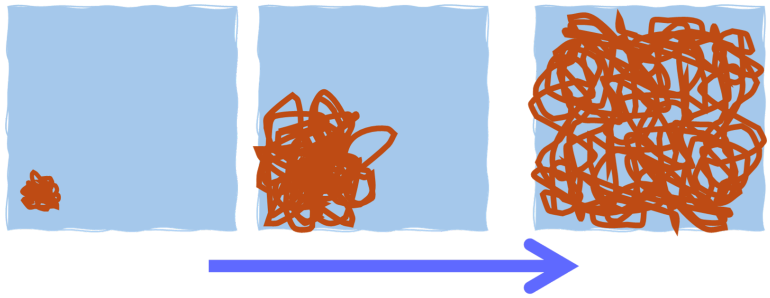

Figure 7. Illustration depicting the kind of images that a growth prediction model can be trained using.

This work explored use of the UNet-CNN model architecture as a solution to the learning task of predicting R. okamurae presence masks at a next timestep given masks from previous ones.

UNet-CNN is a type of CNN that also comprises elements of an encoder-decoder architecture. It is most known for its suitability for image segmentation tasks (i.e., for its ability to predict masks identifying certain objects in an image like separating a human being from the background, etc.) (Aramendia, 2024). Inherently, a UNet-CNN is solely a spatial model and has no natural understanding of sequences. However, one idea amongst the team was that if the input and output were structured so as to correspond to timestep t and timestep (t + 1) respectively, then it may be that the UNet-CNN could pick up on temporal progression to some degree of success. Although experimental, given its simplicity over building a hybrid model with time-aware architectural components like repeating LSTM blocks as in a Recurrent Neural Network (RNN) for example, alongside image feature extraction blocks, this idea of going with a strong image classifier with 2 time-step (t, t+1) input output data pairs was chosen for implementation in interest of feasibility within the available time.

Once the architecture was decided upon, the next challenge was that of data. This task requires a dataset of presence masks which may be binary matrix representations of images where 1s occupy the portion of surface covered by the algae and all remaining pixels equal 0 (see Figure 8). Such a dataset could not be found and would've required more time than available to create. Hence, this work demonstrates how such a model may be leveraged using **simulated data**.

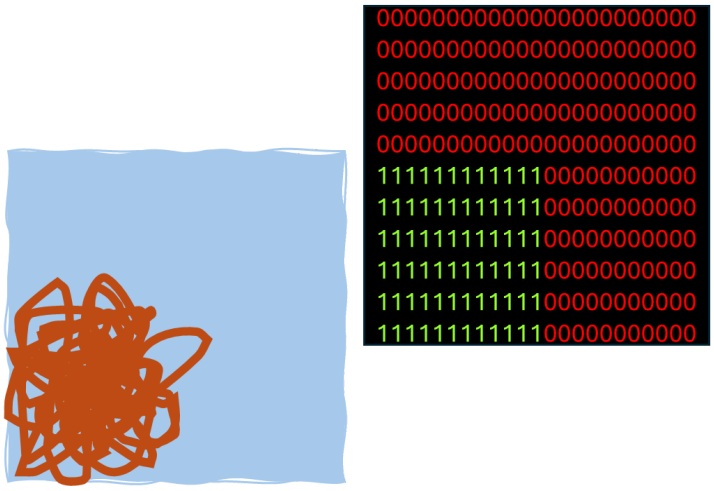

Figure 8. Visual representation of a presence mask matrix.

Synthetic presence mask sequences were generated via a probabilistic approach wherein the first timestep in each sequence is a matrix composed of 0s at all but few random pixels where the value is 1 instead. The 0s and 1s represent absence and presence of the organism respectively in this highly simplified attempt at generating growth pattern data. At each subsequent timestep, pixels surrounding those with a value of 1 may also be set equal to 1 based on some predefined probability. I.e., if this probability is set to be 0.3, then this means that 30% of the time, the value of pixels next to ones with value 1 in timestep t, gets flipped from 0 to 1 in timestep t + 1.

500 sequences composed of 4 timesteps each, were generated wherein each frame is a 64 by 64 pixel binary presence mask that can have anywhere from 1 to 4 colonies with spread probability set to 0.3 (30%). A kernel of size 10 by 10 pixels was used to define the neighbourhood around each pixel at each timestep wherein new presence could appear as per the predefined probability.

These sequences were restructured such that the model could receive a single binary infestation mask and would output a predicted mask representing expected spread at the next timestep. This required that generated sequences be arranged in (x, y) pairs where x = timestep t and y = timestep (t + 1). Figure 9 below, showcases some of the 1500 (x, y) pairs used for training and testing.

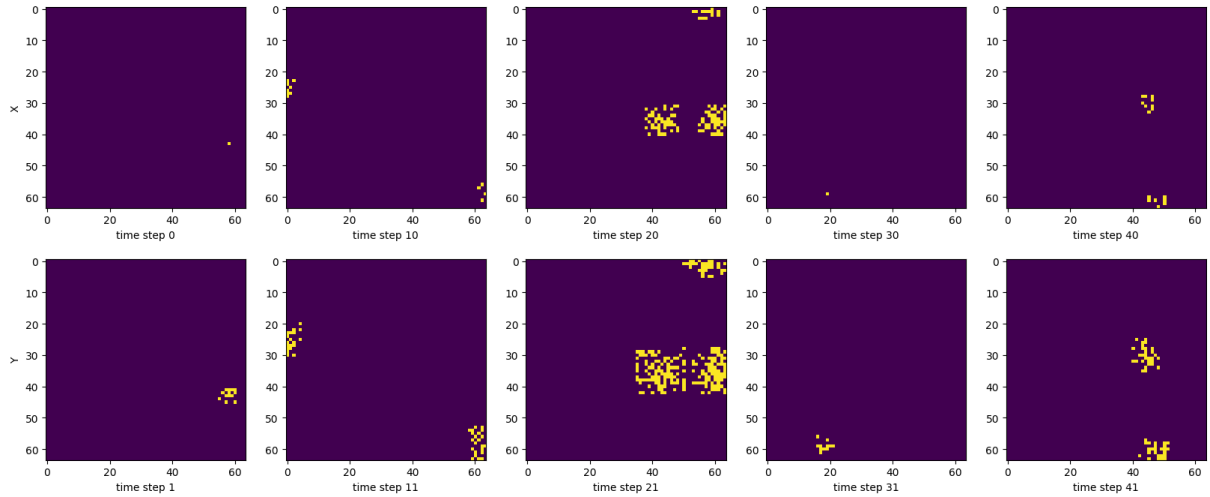

Figure 9. Sample simulated (x, y) infestation growth sequence pairs provided as input to the UNet-CNN model.

The model was trained on 90% of the data (validated on remaining 10%) in batches of 64 images each for 10 epochs using Adam optimizer with BCE loss function. Additionally, the Intersection-over-Union (IoU) metric was used to assess spatial agreement between predicted and true masks. IoU values range from 0 to 1, with 1 indicative of perfect overlap of the predicted mask and the ground truth mask.

Given below are training and validation BCE loss values alongside validation IoU values after each epoch.

```
Epoch 1/10 — train_loss: 0.4617 val_loss: 0.3405 val_iou: 0.0605
Epoch 2/10 — train_loss: 0.3077 val_loss: 0.2509 val_iou: 0.2563
Epoch 3/10 — train_loss: 0.2406 val_loss: 0.2018 val_iou: 0.2623
Epoch 4/10 — train_loss: 0.1901 val_loss: 0.1600 val_iou: 0.2592
Epoch 5/10 — train_loss: 0.1505 val_loss: 0.1320 val_iou: 0.2602
Epoch 6/10 — train_loss: 0.1228 val_loss: 0.1083 val_iou: 0.2589
Epoch 7/10 — train_loss: 0.1025 val_loss: 0.0877 val_iou: 0.2587
Epoch 8/10 — train_loss: 0.0877 val_loss: 0.0761 val_iou: 0.2591
Epoch 9/10 — train_loss: 0.0768 val_loss: 0.0681 val_iou: 0.2591
Epoch 10/10 — train_loss: 0.0688 val_loss: 0.0604 val_iou: 0.2592
Training finished. Best val IoU: 0.26228851238886514
```

It can be observed here, that after epoch 1, although loss continues to reduce, the IoU value does not increase significantly beyond about 0.25, indicating at most 25 to 26% overlap between prediction and ground truth. This observation, combined with inspection of some predictions alongside input, and ground truth as in Figure 10 below wherein it is clear that the UNet-CNN is perfectly recreating the input model in each case, suggests that while the encoder and decoder capture spatial information very well, they completely fail to capture temporal information.

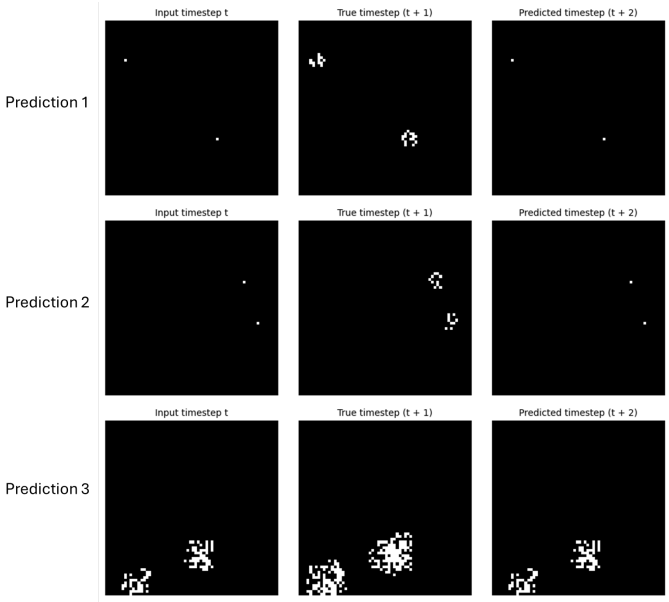

Figure 10. Input at timestep t (leftmost column), ground at timestep (t+1), and prediction for timestep (t+1) of the trained UNet-CNN model for 3 sample inputs from the test set.

This experiment having ended in the model failing to learn, possibly aggravated due to randomness in simulated data generation, is suggestive of the idea of simply restructuring data into a (timestep t, timestep t+1) format to induce temporal prediction being flawed and insufficient for useful performance even when leveraging a strong spatial feature extractor. This result encourages recommendation of an albeit slightly more complex, but much more likely to succeed, hybrid model that incorporates longer (> 2 timesteps) sequential time-aware design decisions (e.g. repeating blocks in RNNs, position encoding in Transformer, autoregressive sequential prediction, etc.) alongside strong spatial feature extractor components to arrive at a practical *R. okamurae* infestation progression predictor model. An example of such an architecture that is potentially more suitable, would be LSTM-CNN. That said, the fact that the model only had 2 timesteps to work with and the significant randomness (no. of initial presence seeds and random neighbouring points being flipped from 0 to 1 with a 30% chance) in simulated sequence generation cannot be discounted as obstacles to learning here. Thus, even with the hybrid model, it is very important to invest in putting together a good-quality dataset containing several sequences of snapshots showing growth of R. okamura over several time steps at different locations with each image having an associated binary presence mask.

Lastly, although here, BCE loss was chosen since masks are binary, it is possible that given their matrix structure and IoU being the other metric to improve, other image-specific loss functions like BCE + Dice loss may drive better weight updates.

Please find all UNet-CNN related code in the `UNetCNN` folder on GitHub.

ROST Early Detection - Temporal GNN

So far, solutions discussed are limited to working with input matrices wherein spatial information is present in the form of value (or pixel in case of images) proximity in the input alone. These inputs do not inherently capture properties that drive propagation such as ocean currents, marine traffic etc, in a scalable fashion.

A more complete solution to early detection of the spread of *R. okamurae* over time along coastlines would require a spatio-temporal model that works with inputs containing both habitat conditions at individual locations and properties of propagation channels that connect them. To this end, this work proposes use of a Temporal Graph Neural Network (Temporal GNN) which due to their ability to learn properties of nodes and links of a graph data structure can capture both environmental conditions as node features and propagation pathway properties as link features.
It was not possible to implement such a model in code in the given timeframe, since just identifying this approach as suitable for *R. okamurae* early detection, was in and of itself, challenging. Other fields were explored to identify a model type that was a good fit for the underlying problem of ***spatio-temporal propagation prediction***. Any temporal variant of the spatially structured GNN model was deemed the best choice due to evidence of its successful application w.r.t solving similar problems in other areas such as cancer metastasis prediction (Agyekum et al., 2025) (Kazmierski & Haibe-Kains, 2021) and epidemiology to predict spread of infectious diseases (Alfas et al., 2025).

Leveraging a temporal GNN model requires relevant data to be structured as a graph across timesteps. The next challenge was thus, to propose a meaningful way to do it and this work suggests the following.

- Let coastlines of the world be divided into fixed length L (say, 100 km) stretches.
- Let each stretch be composed of N (say, 50) equidistant points. Each such (longitude, latitude) coordinate shall be one node in the graph.
- Environmental conditions at each node shall define the vector of node features. Based on literature exploring the spread of R. okamurae (Herrero et al., 2023) (Laamraoui et al., 2024) (Román & Vázquez, 2025) (Haro et al., 2024) (Díaz-Tapia et al., 2025) , these may be *sea surface temperature*, *water depth*, *is substrate rocky*, *biodiversity*, *water salinity*, *water pH*, *nitrogen content*, *oxygen content*, etc.
- Each node shall also be associated with a target variable (the feature that is to be predicted), *is algae present* or *algae cover %*.
- The links in the graph shall capture geographic proximity of location nodes with

- link features comprising properties related to propagation dynamics such as *net direction and magnitude of ocean currents, wind*, *anthropomorphic and nature marine traffic*, etc.
- Thus, a single timestep of a single stretch of coastline where *R. okamurae* was detected, can be represented using an acyclic graph.
- Each data point used to train the model shall contain one graph of the same coastline stretch across T timesteps (say, 20 timesteps with interval between them being 2 weeks) such that the set of timesteps incorporate evolving growth of the algae along the coastline.

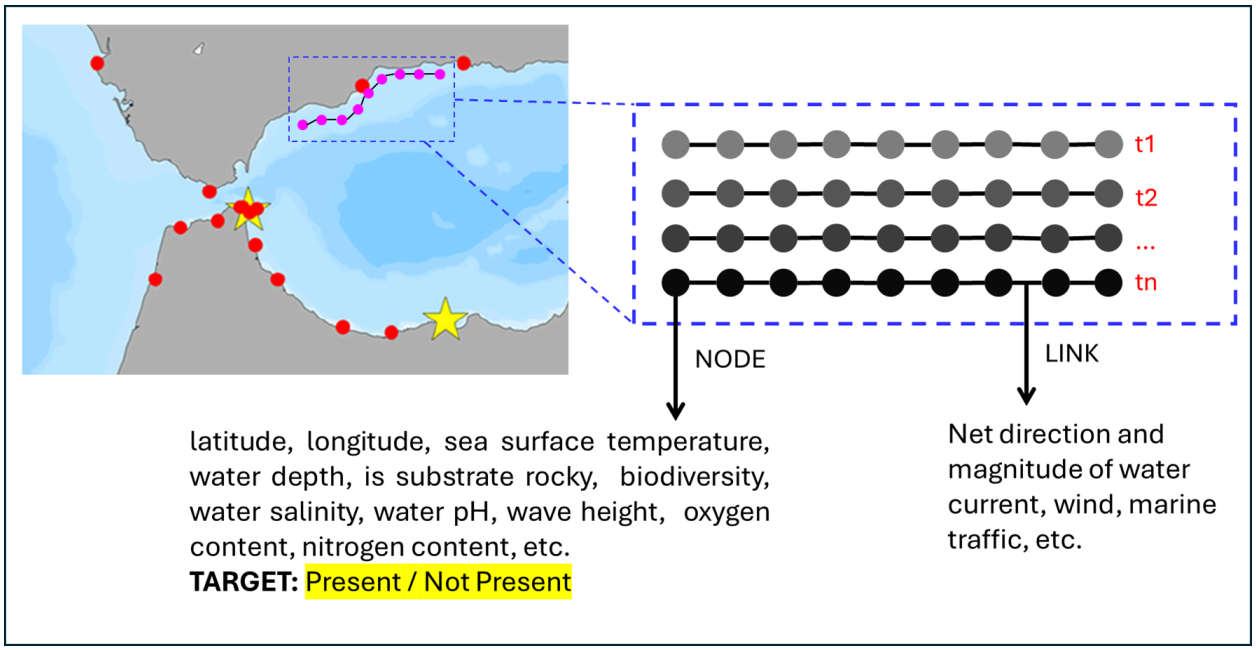

Figure 11. Illustration of proposed graph-ification of information relevant to *R. okamurae* early detection using a temporal GNN. Top left image of coastline with observed points of algae presence was obtained from (Román & Vázquez, 2025).

The training task here would be that of node prediction where the GNN, given past timesteps of a graph, learns to predict the target variable's value at subsequent timesteps, for several graph sequences.

Once trained, the temporal GNN may be provided with the snapshot of a new stretch of unseen coastline at timestep T_0 and made to predict subsequent timesteps such that changing values of the target variable per node per timestep provides a prediction of plausible growth and spread of *R. okamurae* along this new stretch of coastline over time.

GNN Scenarios - Practicalities of Implementation

To support spatial modelling and prediction, two main data types are required: **presence data** and **environmental data**, each providing key inputs for network-based analysis.

Presence data (Nodes)

- *Current:* Presence/absence with location and time
- *Potential addition:* Abundance or coverage index (e.g., low/medium/high)

Environmental data (Nodes)

- *Water temperature:* Growth and beachings peak in warm seasons; first major events align with warm anomalies (e.g., 2015). Broad tolerance ~10–30 °C, with peak growth when max temps > 15 °C — supports including SST and air temperature.
- *Wind direction and speed:* Determine biomass accumulation; significant relationships with coverage/NDVI at 5-day lags.
- *Nutrients:* Higher nutrient levels enhance photosynthesis and growth.

Linked data to connect sampled locations (edges)

- *Currents:* Net direction and velocity are critical for spread and beaching dynamics.
- *Marine traffic/port proximity (edges):* Ports and aquaculture act as introduction points; include a "port proximity" risk layer.

Strategy	Data Source	Accuracy	Cost	Pros	Cons
1. Outsourced	Outsourced presence and environmental data.	Medium	$	No physical sampling necessary.	Satellite data less accurate early detection. Dependent on cooperating with multiple external partners.
2. Internal Sampling	Internal sampling using eDNA.	Medium	$$	Accurate presence data. Work autonomously.	Rely on patterns in biodiversity (eDNA) for predictability. Physical sampling required.
3. Hybrid	Internal sampling using eDNA . Environmental data from databases.	High	$$$	Reliable presence and environmental data. No masking necessary.	Dependent on external partners. Physical sampling required.

Table 2. Accuracy - cost estimates of 3 pathways towards implementing proposed ROST GNN model.

Conclusion

This work investigated several potential solutions to detection / early detection of *Rugulopteryx okamurae*, a species of brown algae that has turned highly invasive in the Mediterranean Sea and Atlantic Ocean and continues to spread, with a focus on Data Science and ML based approaches using open source data. Lack of quality datasets labelled with both presence and absence of the species was a notable limitation that is also indicative of an urgent need for curation of such data to facilitate meaningful use of great predictive insight possible using advanced data-driven AI models.

Correlation-based analysis of occurrence data from GBIF coupled with oceanographic data from the Bio-Oracle Database demonstrated how habitats may be ranked according to suitability for *R. okamurae* based on environmental similarity of locations to geo-coordinates where the species is known to have been found. This approach can further be applied to more data and better correlated features to identify key global locations for active monitoring and concentration of preventive efforts.

Next, data from Copernicus Sentinel-3 Ocean and Land Colour Instrument tagged with simple presence / absence labels using occurrence data from GBIF was used to successfully train a CNN for binary classification of presence 1 or absence 0 of *R. okamurae* from satellite images. Although perfect performance of the trained model on test data (100% accuracy and F1 Score = 1.0) here, is more due to the severely limited and class-imbalanced nature of the dataset, this result alongside observed trends in learning curves lends good evidence to the approach being sound, and detection of the organism from satellite images being indeed possible.

As a step-up in complexity and utility in comparison to simple binary presence detection at given locations at a given time, the idea of predicting changes in percentage cover of the organism over an area over time was entertained. A most simple (most feasible in given timeframe) way to try and do this was thought to be to use a model architecture known for its superior performance in image segmentation with inputs and outputs restructured to be sequential in nature. Since a dataset containing sequences of satellite images labelled with *R. okamurae* presence masks could not be found, a simple dataset with probabilistically generated sequences of masks (all 0 matrices with few 1s indicative of presence) was created to use with the model as synthetic data that could later be substituted with true observations. Synthetic sequences were split into (timestep t, timestep t + 1) pairs and the model was trained on these. The resulting model however, failed to demonstrate useful learning as despite a drop in BCE loss, the performance metric IoU (indicative of how well predicted timestep (t + 1) mask overlaps the true timestep

(t + 1) mask) remained almost the same for the test set across all epochs without large improvements after epoch 1. This reveals that, whilst simpler and faster than introducing recurring blocks (maybe even advanced ones like LSTMs and GRU cells) allowing for sequential (longer than 2 timesteps) inputs as in RNNs or implementing positional temporal encoding as in transformers, simply structuring input and output to be sequential pairs alone is not sufficient for a model to be able to capture temporal trends in data. The UNet-CNN, here, ended up perfectly reproducing spatial patterns. This is not surprising given their known prowess in image segmentation. But it completely failed to capture the temporal part of the assignment. This exercise, although only with synthetic data, drives the recommendation that tasks such as prediction of *R. okamurae* presence masks in images over time, using artificial neural networks, employ hybrid models (e.g. LSTM-CNN) with both image feature extraction components as well as architectural design elements designed specifically for use with longer than 2 timestep sequences of input data.

Lastly, inspired from its successful application in other fields like cancer metastasis prediction from medical images and prediction of the progression of infectious diseases, this work presents a robust and more complete solution to true early detection *R. okamurae* progression along coastlines. This proposed *Rugulopteryx okamurae* Spatio-Temporal (ROST) GNN approach advises modelling stretches of coastline as acyclic graphs where environmental variable values as well as the target presence variable comprise node features, and organism propagation relevant features like water currents and marine traffic, as identified from existing research, define link features. This model trained on several sequences of graphs over multiple timesteps should be able to pick up on spatio-temporal progression patterns of the organism such that given a fresh coastline graph, a trained ROST GNN auto-regressively (output at step t becomes input for step (t + 1)) predicts subsequent values of the target variable associated with each node with acceptable accuracy, thereby effectively predicting growth and spread of *R. okamurae* in a new area.

Although it could not be implemented in code due to data and time constraints, the ROST GNN model idea, grounded in theory, among others explored in this study, was deemed the most complete solution to the problem of *R. okamurae* early detection. Hence, a quick attempt at exploring 3 different pathways to implementing this idea, with prime focus being on the cost and accuracy of each one, in light of the type and amount of data to be fetched and prepared (Table 2) concludes this work.

References

Agyekum, E. A., Kong, W., Ren, Y., Issaka, E., Baffoe, J., Xian, W., Tan, G., Xiong, C., Wang, Z., Qian, X., & Shen, X. (2025). A comparative analysis of three graph neural network models for predicting axillary lymph node metastasis in early-stage breast cancer. *Nature Scientific Reports*, *15*(13918). https://doi.org/10.1038/s41598-025-97257-z

Alfas, M., Kumar, M., Shriyam, S., & Kumar, S. (2025, July). An Efficient Framework for Epidemiological Parameter Estimation via Graph Reduction and Graph Neural Networks. *Association for Computing Machinery*, *19*(6), 1556-4681. 10.1145/3736727

Aramendia, A. I. (2024, January 31). *The U-Net : A Complete Guide*. Medium. Retrieved November 10, 2025, from https://medium.com/@alejandro.itoaramendia/decoding-the-u-net-a-complete-guide-810b1c6d56d8

Díaz-Tapia, P., Alvite, N., Bañón, R., Barreiro, R., Barrientos, S., Bustamante, M., Carrasco, S., Cremades, J., Iglesias, S., Rodríguez, M. d. C. L., Muguerza, N., Piñeiro-Corbeira, C., Quintano, E., Tajadura, F. J., & Díez, I. (2025, January). Multiple introduction events expand the range of the invasive brown alga Rugulopteryx okamurae to northern Spain. *Elsevier Aquatic Botany*, *196*(103830). https://doi.org/10.1016/j.aquabot.2024.103830

Haro, S., Morrison, L., Caballero, I., Figueroa, F. L., Korbee, N., Navarro, G., & Bermejo, R. (2024, June). Understanding the invasion of the macroalga Rugulopteryx okamurae (Ochrophyta) in the northern Alboran Sea through the use of biogeographic models. *MDPI Remote Sensing*, *16*(15). 10.3390/rs16152689

Herrero, J. J., Simes, D. C., Abecasis, R., Relvas, P., Garel, E., Martins, P. V., & Santos, R. (2023). Monitoring invasive macroalgae in southern Portugal: drivers and citizen science contribution. *Frontiers in Environmental Science*. 10.3389/fenvs.2023.1324600

Kazmierski, M., & Haibe-Kains, B. (2021). Lymph Node Graph Neural Networks for Cancer Metastasis Prediction. *arXiv e-prints*. 10.48550/arXiv.2106.01711

Laamraoui, M. R., Mghili, B., Roca, M., Chaieb, O., Ostal´e-Valriberas, E., Martín-Zorrillae, A., Sabino-Lorenzo, A., & Aarab, S. (2024). Rapid invasion and expansion of the invasive macroalgae Rugulopteryx okamurae in the Mediterranean and Atlantic: A 10-year review. *Marine Pollution Bulletin*, *209*. https://doi.org/10.1016/j.marpolbul.2024.117194

Román, S., & Vázquez, R. (2025, May). Assessment of the Rugulopteryx okamurae invasion in Northeastern Atlantic and Mediterranean bioregions: Colonisation status, propagation hypotheses and temperature tolerance thresholds. *Elsevier Marine Environmental Research*, *207*(107093). https://doi.org/10.1016/j.marenvres.2025.107093

Sanderson, G., & 3Blue1Brown. (2020, September 3). *Convolutions in Image Processing | Week 1, lecture 6 | MIT 18.S191 Fall 2020*. YouTube. Retrieved November 8, 2025, from https://www.youtube.com/watch?v=8rrHTtUzyZA

Sanderson, G., & 3Blue1Brown. (2022, November 18). *But what is a convolution?* YouTube. Retrieved November 8, 2025, from https://www.youtube.com/watch?v=KuXjwB4LzSA