Marine Environmental Research 216 (2026) 107888

Contents lists available at ScienceDirect

Marine Environmental Research

journal homepage: www.elsevier.com/locate/marenvrev

Harnessing the power of Squidle+ to develop flexible machine learning models

Leonard Günzel [a,*], Jacquomo Monk [b,*], Chris Jackett [c], Ariell Friedman [d], Ashlee Bastiaansen [b], Ardalan Najafi [e], Alberto García Ortiz [e], Neville Barrett [b]

[a] Norwegian University of Science and Technology (NTNU), Marine Technology, Trondheim, Norway
[b] Institute for Marine and Antarctic Studies (IMAS), Hobart, Australia
[c] CSIRO, Hobart, Australia
[d] Greybits Engineering, Australia
[e] University Bremen, ITEM, Bremen, Germany

ARTICLE INFO

Dataset link: https://squidle.org/

Keywords:
Image annotation
Benthic monitoring
Ecological monitoring
Ecklonia radiata
Macroalgal cover
Coral cover
Seagrass cover
Deep learning

ABSTRACT

While our ability to capture seafloor imagery has expanded dramatically, extracting information remains a major bottleneck, as annotation is still largely manual, time-consuming, and costly. Machine Learning (ML) classifiers offer a promising solution, but they often lack generalisation as they are typically trained on small, homogeneous datasets from selected campaigns. Here, we address this challenge by leveraging the Squidle+ platform to assemble a heterogeneous dataset of 1.7 million annotations across 150,000 images from 325 expeditions in Australian waters. To effectively train models on this scale, we introduce a novel dataset distribution strategy, the neighbour-distribution, and implement adaptive cropping to account for varying image resolutions. To assess performance at a national scale, we excluded entire campaigns from training and used them as independent test sets, allowing evaluation on complete transects, in addition to composed datasets. Using this flexible training pipeline, we combined diverse annotation schemes into three high-level Essential Ocean Variables (EOVs): Seagrass, Canopy-Forming Macroalgae, and Hard Coral cover and one species-specific label, *Ecklonia radiata*. This represents the first use of such a large and diverse dataset for training an ML classifier of *Ecklonia radiata*. The resulting models achieved test accuracies of 85%–93% for the EOVs and 96.5% for *Ecklonia radiata* on campaign test sets spanning Australia's coastline, demonstrating robust performance at a national scale. To enhance accessibility, both the code and trained models are made publicly available and directly integrated into Squidle+, lowering the barrier for their use in ecological monitoring programs.

1. Introduction

Underwater imaging has emerged as an essential non-destructive tool for understanding the transformations occurring in the oceans and studying their diverse aspects (Katija et al., 2022). Over time, this technology has made remarkable advancements in terms of quantity, quality, and spatial coverage (Schoening et al., 2022). The increasing widespread deployment of autonomous and remote underwater vehicles (AUVs and ROVs) has played a significant role in expanding the spatial coverage of underwater imaging. As a result, there has been a substantial upsurge in the amount of research focusing on marine imagery over the past three decades, with reports indicating a one-hundred-fold increase (Durden et al., 2016). To extract the data contained in these images, they need to be annotated, which is the process of labelling objects of interest captured within the imagery to derive physical, biological, and/or ecological data (Gomes-Pereira et al., 2016).

In practice, point-count annotation methods are by far the most widely used in benthic imagery analysis, either as targeted points manually assigned to features of interest or as random/grid points superimposed on imagery. Studies such as Perkins et al. (2016), Taormina et al. (2020), and Perkins et al. (2022) have demonstrated the importance of point-based annotation in providing repeatable, scalable, and statistically robust measures of benthic community composition across different habitats and monitoring programs. For a point-count approach, a defined number of points are digitally superimposed across each media (e.g. an image) across a subset taken from deployment (e.g. a transect). The annotator can then identify and assign a label to

* Corresponding authors.
E-mail addresses: leonard.gunzel@ntnu.no (L. Günzel), jacquomo.monk@utas.edu.au (J. Monk).

https://doi.org/10.1016/j.marenvres.2026.107888
Received 15 November 2024; Received in revised form 15 January 2026; Accepted 26 January 2026
Available online 9 February 2026

the object or organism where the point is located. For more information, we refer the reader to the Australian Field Manual (Monk et al., 2024).

The annotation process remains a major bottleneck in effectively harnessing the extensive volume of image data available, which is primarily reliant on human annotators and proves to be both time-consuming and costly (Gaston and O'Neill, 2004; MacLeod et al., 2010; Aguzzi et al., 2011). To overcome this bottleneck, one proposed method is to use Machine Learning (ML) models to annotate data (Purser et al., 2009; Moniruzzaman et al., 2017). The implementation and training of robust ML models for image annotation have become more accessible with frameworks like PyTorch (Paszke et al., 2019) and TensorFlow (Abadi et al., 2015), which allow users to easily train their neural networks or repurpose pre-trained models. However, despite these advancements, training models from scratch typically requires large datasets, and even transfer learning approaches, while capable of achieving reasonable performance with fewer examples, generally require substantial amounts of diverse data to ensure robust performance and generalisation across conditions (Hernandez et al., 2021).

Access to such data is often limited to what was collected during specific campaigns or longer time series, which are not always publicly available. Additionally, differences in data formatting between institutions or platforms create further barriers to utilising the data efficiently. As a result, access to large and heterogeneous datasets remains constrained (Schoening et al., 2022). Consequently, previous research has highlighted a significant challenge: models trained on data from a restricted set of expeditions show reduced reliability when making predictions on new, unseen data (Katija et al., 2022).

To address this challenge, we present an approach for crowdsourcing a large, heterogeneous training dataset of labelled point-count annotation data, utilising the online annotation and data-sharing platform Squidle+ (Friedman et al., 2026). Squidle+ is one of the largest repositories of openly accessible and easily discoverable georeferenced marine images with associated annotations. In this paper, we present an ML model that has been trained on a large dataset from over 325 expeditions, comprising 150,000 images and 1.7 million annotations from Australian waters. The dataset comprises several individual datasets that have been captured using different imaging platforms (AUV, ROV, Diver) at a range of depths and lighting conditions. Implementing such a diverse dataset as training data introduces significant challenges related to data distribution and quality.

In this work, we discuss three different data distribution methods optimised for training ML models on point-count annotation, one of which is the novel neighbour-distribution method. We demonstrate the effectiveness of our pipeline by developing four distinct ML models capable of annotating the Essential Ocean Variables (EOVs): Seagrass, Canopy-Forming Macroalgae, and Hard Coral cover, as defined by the National Marine Science Committee (Hedge et al., 2022) and the Global Ocean Observing System (Moltmann et al., 2019). Additionally, at the species level, we focused on the kelp *Ecklonia radiata* (referred to hereafter as *Ecklonia*), which is of particular interest in temperate reefs around Australia. The selected EOVs were chosen for their significance and feasibility, as determined by these institutions. They provide valuable insights into climate, ocean ecosystems, and ocean health while being technically, politically, and economically feasible to monitor (Hedge et al., 2022; Moltmann et al., 2019). Kelp forests, often dominated by *Ecklonia*, serve as key habitats for numerous species in Australia's sub-tropical and temperate regions. As such, kelp coverage can be used as an important indicator of ecosystem health and biodiversity (Krumhansl et al., 2016).

In addition to dataset distribution strategies, we also introduce an adaptive cropping approach that defines patch size as a proportion of the original image dimensions rather than a fixed pixel value. This method accommodates the heterogeneity of image resolutions present in large, multi-campaign datasets and ensures more consistent representation of annotated features across varying spatial scales.

Moreover, we address the challenge of bridging the gap between manual and automated annotation, particularly the barriers to adopting newly developed algorithms, by implementing all presented models in the user interface of Squidle+. While extensive research has explored a wide range of models, many remain difficult to access due to limited availability and lack of integration into public annotation tools. As a result, biological experts without a programming background may find it challenging to fully benefit from these advancements (Richards et al., 2019; Trotter et al., 2025). We envision such models as assistants that take over repetitive annotations, while ecological experts guide their application by selecting species- or object-specific classifiers relevant to the study. Our models were therefore designed as binary classifiers, each targeting a single species of interest, reflecting the fact that experts often know which taxa to expect and can exclude irrelevant classes. This targeted approach reduces the risk of misclassification compared to general multi-class models and allows pre-trained classifiers to be applied efficiently in coverage studies, freeing experts to focus on the more complex or uncertain cases. In this study, we:

1. Develop a flexible method for assembling large, heterogeneous point-annotation datasets via the Squidle+ platform.
2. Introduce the neighbour-distribution strategy to improve model performance and reduce dataset size.
3. Implement adaptive cropping to accommodate varied image resolutions.
4. Train and evaluate four binary classifiers for specific species or Essential Ocean Variables (EOVs), making them publicly available and integrated into Squidle+.

2. Materials and methods

The overall workflow for this study, which also defines the structure of the following sections, is illustrated in Fig. 1. We begin by describing our first contribution, the acquisition of large-scale point-annotation datasets through the Squidle+ platform (Section 2.1). Leveraging this resource, we address the challenge of dataset composition and introduce the novel neighbour-distribution strategy for effective curation (Section 2.2). In the context of dataset composition, we also discuss training dataset composition and the evaluation of such in Section 2.3. Based on the curated dataset, we then examine individual images and annotations to determine the optimal extraction of information using an adaptive cropping approach (Section 2.4). Finally, we present the ML model and architecture, along with the optimiser and data augmentation evaluated in Sections 2.5 and 2.6.

2.1. Data acquisition via Squidle+

Squidle+ is a software platform for the management, discovery, and annotation of marine imagery, enabling users to link image data and create structured annotation sets. Alongside Squidle+, platforms such as FathomNet (Katija et al., 2022), CoralNet (Chen et al., 2021), ReefCloud (Gonzalez-Rivero et al., 2025), and BIIGLE (Langenkämper et al., 2017) play a central role in providing large, heterogeneous annotation datasets and supporting community benchmarking efforts.

Across these platforms, annotations can be generated in multiple modes, including whole-frame, point-based, and polygon-based annotations, and applied to diverse media types such as images, videos, and mosaics. In this study, we restrict our analysis to data managed within Squidle+ and focus exclusively on point-based annotations on still images. However, the procedures and workflows described here be readily extended to other annotation platforms that support comparable data structures and annotation paradigms

On the Squidle+ platform, each annotation entry is a table row containing information such as the annotation label, the reference to the corresponding image, and the coordinates of the annotation point within that image. Since multiple annotations can refer to different

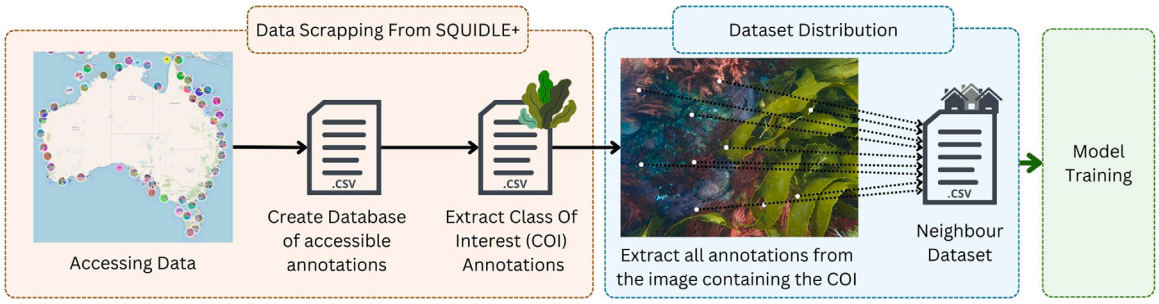


Fig. 1. Workflow for dataset creation and distribution with neighbour condition. Annotation data are accessed via the Squidle+ platform and compiled into a database of available annotations. From this, only the Class of Interest (COI) annotations are extracted. For each COI, all co-occurring annotations from the same image are included to form the neighbour dataset, which is then used for model training.

points within the same image, we first exported all accessible entries into a CSV file containing the annotation label, image reference, and annotation coordinates (see the full API call in the GitHub repository (Günzel, 2023)). From this dataset, we obtained 61,100 *Ecklonia* annotations from 17,000 images; 6800 Seagrass cover annotations from 2300 images; 202,300 Canopy-Forming Macroalgae cover annotations from 33,000 images; and 84,300 Hard Coral cover annotations from 30,000 images.

The models presented in this study were developed with open data accessibility as a guiding principle. For *Ecklonia*, 97.2% of the target-class annotations used for model training are publicly available; for Seagrass, 96.7% of the training annotations are publicly available; and for Canopy-Forming Macroalgae, 95.5% of the training annotations are publicly available. To increase the overall annotation count for Hard coral and ensure sufficient class representation, we requested access to additional annotation repositories beyond those that are publicly available. As a result, 49.7% of the target-class annotations used for training Hard coral are publicly accessible.

A comprehensive overview of all datasets used in this study, including their access status (public or private), is provided in Appendix C.3. While some datasets are currently not publicly accessible through the Squidle+ platform, data owners may grant access for model development upon request, and several datasets classified as non-public are expected to become publicly accessible as contributing researchers complete ongoing analyses. The datasets analysed in this study reflect the state of the platform in April 2023; since that time, a substantial number of additional campaigns have been added, meaning that contemporary models trained exclusively on publicly available Squidle+ data are likely to have access to a larger and more diverse training corpus than was available for the models presented here.

Once the annotationsets are compiled we proceed with the final training distributions, certain annotation entries were systematically excluded as outlined below. In all cases, annotations flagged as "experimental" were removed, along with annotations belonging to taxonomic levels higher than our Classes Of Interest (COI). This step was necessary due to inconsistencies in taxonomic detail used across datasets (Friedman et al., 2026). For example, in our study, it is important to distinguish between the macroalgae *Ecklonia* and *Phyllospora comosa*, whereas another study annotating at higher taxonomic layer might generalise both under Canopy-Forming Macroalgae. In such cases, an image showing *Ecklonia* but annotated only at the higher taxonomic layer as "Canopy-Forming Macroalgae" would be treated by our model as a non-target class and thus would obscure the training process.

Once the annotation entries are attributed to the datasets, the images that the annotation entries refer to are checked for redundancy and downloaded. As the datasets we are working with are crowdsourced, we are presented with very heterogeneous collections of images (such as different depths, locations, lighting, camera systems). This heterogeneity presents challenges as we aim for minimal individual data preprocessing (like manual optical examination of the data), as this would make our workflow inapplicable to other COI. Thus, we omitted manual preprocessing of the data, even though the literature has shown this to be a promising step to improve the performance of the model (Jackett et al., 2023).

2.2. Dataset composition

Out of the 1.7 million annotations, approximately 61,100 annotations (~4%) are labelled as *Ecklonia*. Thus, training on the complete dataset would imply a target-to-non-target annotations ratio of 1:27. To avoid excessive training time due to the large non-target dataset, we crafted three smaller training datasets with varying annotation distributions for comparison.

The first approach taken was to create a dataset with a target-to-non-target annotation ratio of **1:1**. Therefore, 62,000 *Ecklonia* annotations were matched with 62,000 randomly selected "Others" annotations using stratified sampling. Stratified sampling is the process of randomly sampling a small dataset from a larger one while maintaining the same proportion of classes as in the larger dataset. Thus, if the label "coarse sand (with shell fragments)" made up 39.6% of entries in the original dataset, it would make up 39.6% in the Others category (as displayed in Fig. 2).

For the second dataset, we selected a ratio of **1:10**. The distribution of labels stayed the same (as shown in Fig. 2), but the amount of "Others" annotations was increased tenfold, while the amount of Ecklonia annotations stayed the same. This dataset required significantly more computer resources to train on, and so demanded more computing time. Thus, we looked for ways to achieve an adequate label distribution according to our computing resources, while simultaneously maintaining or potentially improving the model's performance.

We came up with a third dataset distribution, which we call the **neighbour distribution**. In the case of this dataset, all annotations from a single image would be taken into account provided that there was at least one *Ecklonia* annotation connected to that image (see Fig. 1). Therefore, the dataset was focused on *Ecklonia* and species that are commonly spatially close to *Ecklonia*. We further added some common classes, as described for the 1:1 and 1:10 ratio dataset, to this dataset to incorporate common, but less spatially close species, which resulted in a dataset size similar to the 1:1 ratio. The resulting dataset showed a more equally distributed coverage of labels. As a result, we could observe the presence of classes like the canopy-forming macroalgae, *Phyllospora comosa*, in the top eight most common classes, that were not present in either the 1:1 or 1:10 ratio (see Fig. 2 right depiction).

2.3. Evaluation

The dataset that the model is trained on is divided into two smaller excerpts: the training and validation datasets. In our case, we split 80% of the data for training and 20% for validation. Before creating the

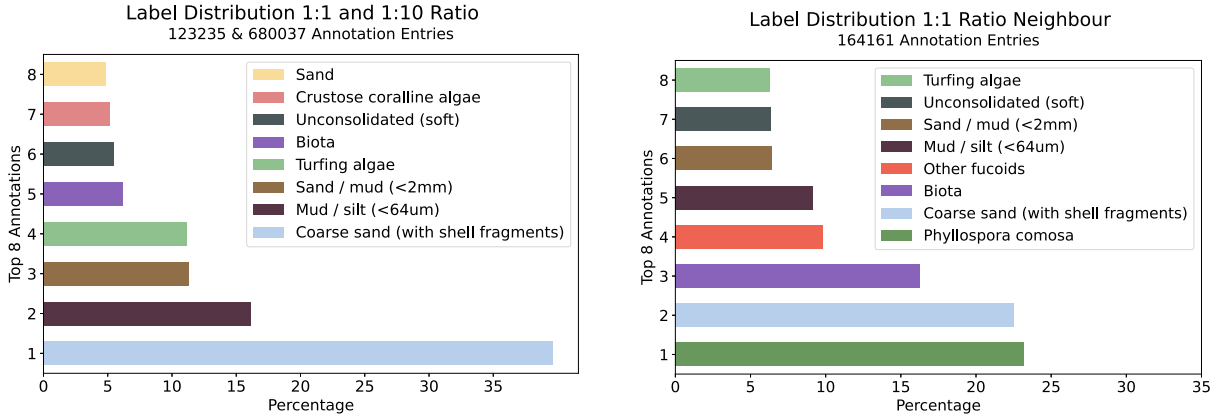


Fig. 2. Distribution of the top eight annotations by percentage. The left chart combines the 1:1 and 1:10 ratios, as they share the same distribution percentage and differ only in the total number of annotations. The right chart shows the neighbour distributed dataset with approximately the same amount of annotations as the 1:1 dataset. Introducing the neighbour condition increased the overall class balance and increased the representation of species commonly found in close proximity to *Ecklonia*, such as *Phyllospora comosa*.

training and validation sets, we excluded three random campaigns from the whole dataset for each EOV, and for *Ecklonia*, we excluded five additional campaigns from different states in Australia, to allow the evaluation of the model generalisation properties at a national scale.

These excluded campaign datasets serve as a control group, allowing us to assess model performance on complete campaigns. We tested the *Ecklonia* model on eight test campaigns with 1097 to 2501 annotations each and a coverage per dataset ranging from 8.5% to 36.7% (see Additional Information Table C.11). A similar number of entries and percentages were also achieved for the EOV test sets; additional information on these test sets can be found in Additional Information Table C.12. This provides a practical benchmark against the validation dataset, which consists of a random mixture of cropped patches from multiple transects. Thus, in the coming sections, we distinguish further between validation and campaign test results.

Further, we incorporate a 5-fold cross-validation for the final evaluation step of each model. After splitting the dataset into training and validation, there is a chance that the distribution is favourable for our model (e.g. most simple annotation data are used for validation). Thus, the dataset is split up into 5 equally sized parts, four of which make up the training and one is allocated as validation data. The allocation of the subsets (training or validation) is switched for every iteration of the training.

To evaluate the performance of our model, we will focus primarily on the F1-score. For completeness, we will also report accuracy in the final results. For more information and background on these metrics, we refer the reader to Appendix A.2.

2.4. Adaptive cropping

Each annotation marks a specific point within an image. When placing such a point, the annotator relied not only on the immediate pixel but also on the surrounding visual context to identify the species correctly. To reproduce this context for model training, we treat the annotation point as the centre of a square patch extracted from the image. The challenge lies in selecting an appropriate patch size: it must be large enough to capture the relevant features and context, yet small enough to avoid introducing excessive noise from other species in the vicinity.

All models of *Ecklonia* classifiers analysed in the literature covered in this publication utilised a fixed patch size defined by a pixel value (Mahmood et al., 2016a; Beijbom et al., 2016; Stokes and Deane, 2009; Mahmood et al., 2016b; D'Archino et al., 2021; Bewley et al., 2012; Denuelle and Dunbabin, 2010). The size of the patch is therefore defined with a set pixel size. Many of these studies used data from one expedition or one single platform. Thus, their image data is homogeneous in size. Our final training dataset was made up of 198 different image sizes, ranging from 640x480 pixels up to 7,360x4,912 pixels. A fixed patch size of 300x300 pixels could comprise anything between 0.25% to 29.29% of the relative image area, and therefore result in highly variable scaled image content in the cropped patches. Even if larger patches were selected, this would not mean an increase in resolution but rather in perspective, because the extracted patch from an image always has to be scaled to a defined resolution to be compatible with the fixed layer size of the model (in our case, 299 by 299 pixels), which the model can be trained on as illustrated in Fig. 3.

To address this problem, we defined the patch size as a percentage of the image size and therefore achieved more normalised observations. The patch size is here defined as a percentage of the image as given by the following equation:

$$BoundingBox = \frac{Width + Height}{2} \times CropPercentage \quad (1)$$

Another challenge arose when selected points were near the image border, causing the resulting patch to contain empty values. While a common solution is to exclude such data from training, we incorporated it by using the resize function from Python's torchvision library, which applies interpolation to scale images and fill the entire square patch as shown in Fig. 3. Interpolation estimates colour values for new pixels inserted during resizing. This issue becomes more frequent with larger patch sizes, increasing the need for scaling.

Further, we expect different species to require different patch sizes. For example, a spatially extensive species such as *Ecklonia* may occupy a larger portion of an image compared to a seagrass blade perceived from the same distance. To determine the most suitable patch size, we performed a grid search ranging from 1%–100% in 5% increments. Once a peak was identified, we refined the search using 2% increments to determine the optimal size more precisely.

2.5. Model architecture

The extracted patches were used to train a Convolutional Neural Network (CNN) as illustrated in Fig. 4. While the field of computer vision has advanced beyond CNNs to newer approaches, CNNs remain a well-established choice, supported by robust tools and infrastructure for further development and deployment. For a more detailed description of CNNs and ML models in general, we refer the reader to Appendix A.1.

The field has seen many different CNN architectures emerge and engage in standardised competitions like the ImageNet Large Scale Visual Recognition Challenge (Deng et al., 2009). These models are

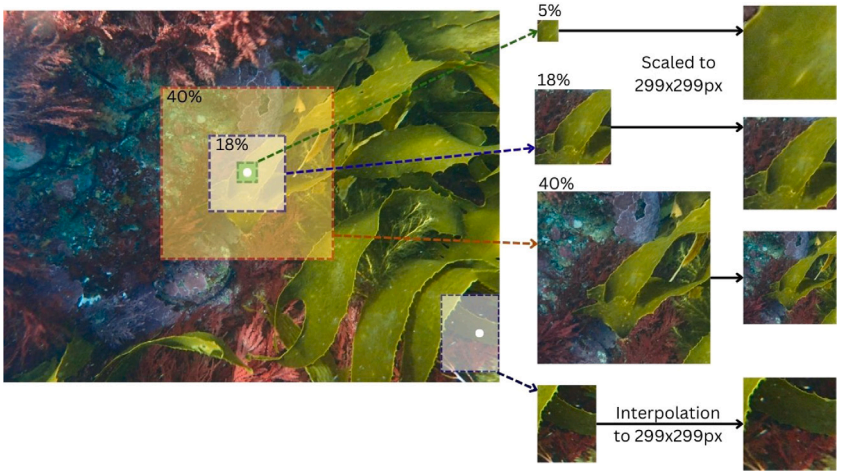


Fig. 3. Illustration of the adaptive cropping approach. Annotation points are placed at the centre of square patches, defined as a percentage of the original image size (e.g., 5%, 18%, 40%). All patches are resized to a fixed resolution of 299 × 299 pixels to enable compatibility with CNN input layers. When annotation points are located near the image border, interpolation is applied to fill empty areas and maintain patch size consistency.

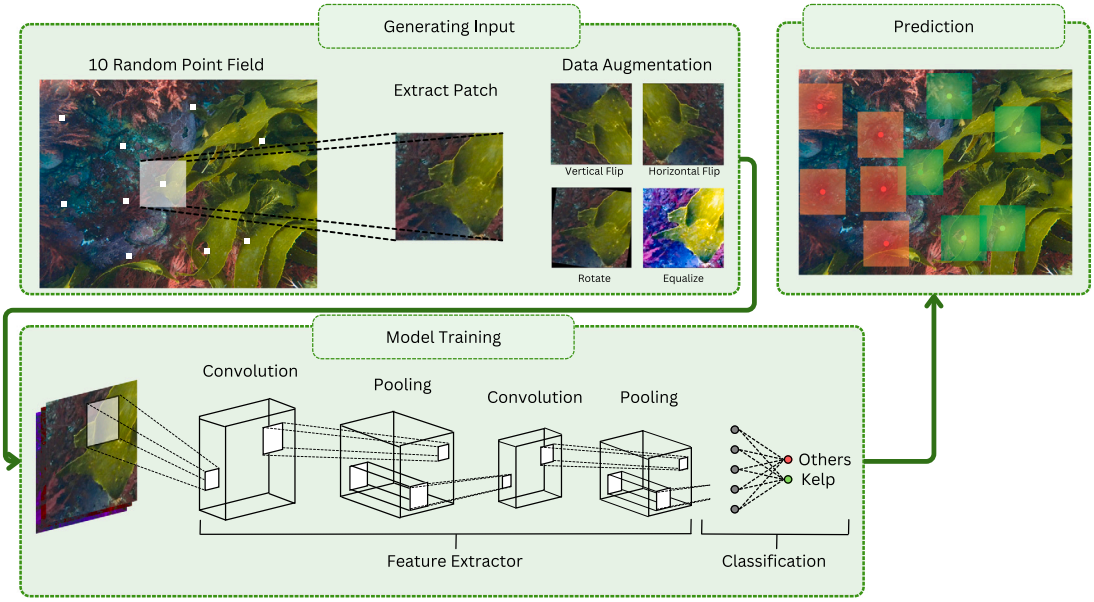


Fig. 4. Illustration of the training process of the model. Out of the 10 random point field, a patch is drawn around one point, and this box is extracted. In the case of training the network, this patch is modified using data augmentation to artificially increase the dataset size and appearance. This step is omitted for prediction. The patch is then fed into the feature extractor, and the output of the feature extractor is classified using the fully connected network at the final layer of the network. The sum of these predictions is illustrated in the final step.

trained on large datasets and are backed by a large amount of computing resources. In this work, we make use of these pre-trained weights, which we adopt and fine-tune through transfer learning, allowing us to leverage state-of-the-art models at a fraction of the original training cost.

Transfer learning refers to the ML method of leveraging knowledge gained from a previously trained model on one task and applying it to a different but related problem. This enables the adaptation of pre-existing models to new contexts, thereby improving efficiency and potentially enhancing performance on the new task (Erhan et al., 2010). This technique is commonly used in CNN-based models as it can speed up the learning process and enhance their ability to generalise (Guo et al., 2016). There are different approaches to initialising and training such a network. While initialising the network, it can be either loaded with the trained weights or randomly initialised weights using the same architecture. To keep the low-level pattern recognition capabilities of the models, loading the trained weights is the most appropriate approach (Chu et al., 2016). In terms of training, there are two alternatives: either freeze some parts of the network and train the remaining ones, or train the entire network through fine-tuning. Despite its high computational cost, fine-tuning was deemed beneficial for this work, as it has proven to yield superior prediction accuracy and generalisation when compared to alternative techniques (Chu et al., 2016). Finally, we replaced the original final layer, which is typically designed for multiclass classification, with a binary classification layer, reducing the output to our two categories: “class of interest” and “other”.

To identify the most suitable model architecture for our use case, we compared nine pre-trained ML models Inception_V3 (Szegedy et al., 2016), Convnext (Liu et al., 2022) (small and large), Regnet (Schneider et al., 2017), Swin (Liu et al., 2021), Resnet (He et al., 2015), Resnext (Xie et al., 2017), Googlenet (Szegedy et al., 2015) and Efficientnet (Tan and Le, 2019). The models were trained on an NVIDIA A40 GPU, and all code was implemented in Python using the PyTorch library.

2.6. Optimiser & augmentation

Model performance can be strongly influenced by the choice of optimiser and the use of data augmentation. In this study, we evaluated five different optimisers and the best combination out of twelve augmentation functions.

Optimisers adjust the learning rate during training, controlling how much model weights are updated and influencing both speed and stability. Many start with a higher rate for rapid convergence before reducing it to fine-tune performance. The most common way to determine the best optimiser and learning rate is through empirical testing; accordingly, we evaluated five widely used options for computer vision tasks (Choi et al., 2019) AdamW (Loshchilov and Hutter, 2019), Adam (Kingma and Ba, 2014), Adamax (Kingma and Ba, 2014), SGD (Sutskever et al., 2013) and RMSprop (Hinton, 2012).

While the dataset is large overall, some classes have few annotations and are represented in only a limited number of campaigns. This narrow exposure can cause overfitting, where the model adapts too closely to the training data and performs poorly on new data. To address this, we applied data augmentation, random manipulations of the original images before training (see Fig. 4), to increase variability. For this study, we limited ourselves to 12 augmentation operations predefined in the PyTorch library. To identify the most effective augmentation strategy, we employed the open-source framework Optuna (Akiba et al., 2019). Optuna uses Bayesian optimisation to efficiently explore parameter combinations while pruning unpromising trials, thereby reducing computation time compared to an exhaustive grid search.

2.7. Deployment on Squidle+

To facilitate the use of the models and encourage uptake within the biological community, especially among experts without a programming background, the best-performing models were integrated into the Squidle+ platform. Through this platform, users can link their datasets and generate point-based annotation datasets. By selecting the models described in this publication, they can automate parts of their workflow without the need to interact with any code. The models are currently limited to point annotations, but the training procedure outlined in this work can also be extended to other forms of annotation, such as full-frame annotation.

3. Results

To achieve the best model performance for our use case, we evaluated design choices including model architecture, dataset composition, crop size, optimiser, and augmentation strategies. We found that transfer learning with fine-tuning of the Inception_V3 architecture, combined with the neighbour distribution dataset and adaptive cropping, provided the best balance of performance and computational efficiency. These optimisation steps resulted in four final species- or EOV-specific models, with F1 scores of 79.0% for Seagrass and Hard Coral, 83.0% for Canopy-Forming Macroalgae, and 91.0% for *Ecklonia*.

3.1. Cropping grid search

The grid search was conducted for all three datasets: 1:1 ratio, 1:10 ratio, and neighbour distribution using the Inception_V3 model, which had emerged early on as the most suitable architecture. In the following, we compare the validation and campaign test results, focusing on trends and alignment rather than specific performance values, as the models are not yet fully adjusted.

For the dataset with a 1:1 ratio, the best campaign test set performance occurred at a crop percentage of 15%, while the highest validation score was achieved at 20% (see Fig. 5(a)). Beyond this point, the gap between validation and campaign test results widened. We would also expect performance to decline after the optimal crop percentage (20%), as larger crops introduce more irrelevant content (“noise”), which can degrade model accuracy. This decline is evident in the campaign test datasets but not in the validation set.

Table 1
Accuracy (Acc.) and F1-Score of different model architectures for Validation and Test runs. Results are highlighted with a colour gradient for better clarity and the test results are averaged over all eight test sets. It can be observed that the Inception_v3 model performs better compared to the other models tested.

	Validation %		Test %	
	Acc	F1-Score	Acc avg	F1-Score avg
Inception v3	**93.2**	**90.8**	**96.1**	**89.9**
Convnext large	89.7	86.2	91.7	77.5
Regnet x 32gf	89.3	85.6	91.4	76.6
Swin v2 b	89.2	85.5	91.3	76.4
Resnet50	88.4	84.4	91.1	74.8
Resnext101 64x4d	88.1	83.9	90.6	74.4
Convnext small	88.0	83.7	91.1	74.8
Googlenet	84.6	79.0	89.0	70.4
Efficientnet v2 l	78.1	69.4	82.0	56.4

The 1:10 dataset shows closer alignment between campaign test and validation results, as well as higher performance on the campaign test set. For both datasets, the best results were achieved at a crop percentage of 30% as shown in Fig. 5(a). However, the larger number of annotations increased the average training time by approximately a factor of four.

The neighbour dataset achieved performance comparable to the 1:10 dataset while using only one-fourth of its size. As shown in Fig. 5(c), it displays strong alignment between campaign test and validation results and good overall performance for both. At very high crop percentages (above 70%), we observe unusual behaviour, likely due to the effects of interpolation and scaling at such large crop sizes. Since this occurs well beyond the optimal crop percentage, it can be disregarded.

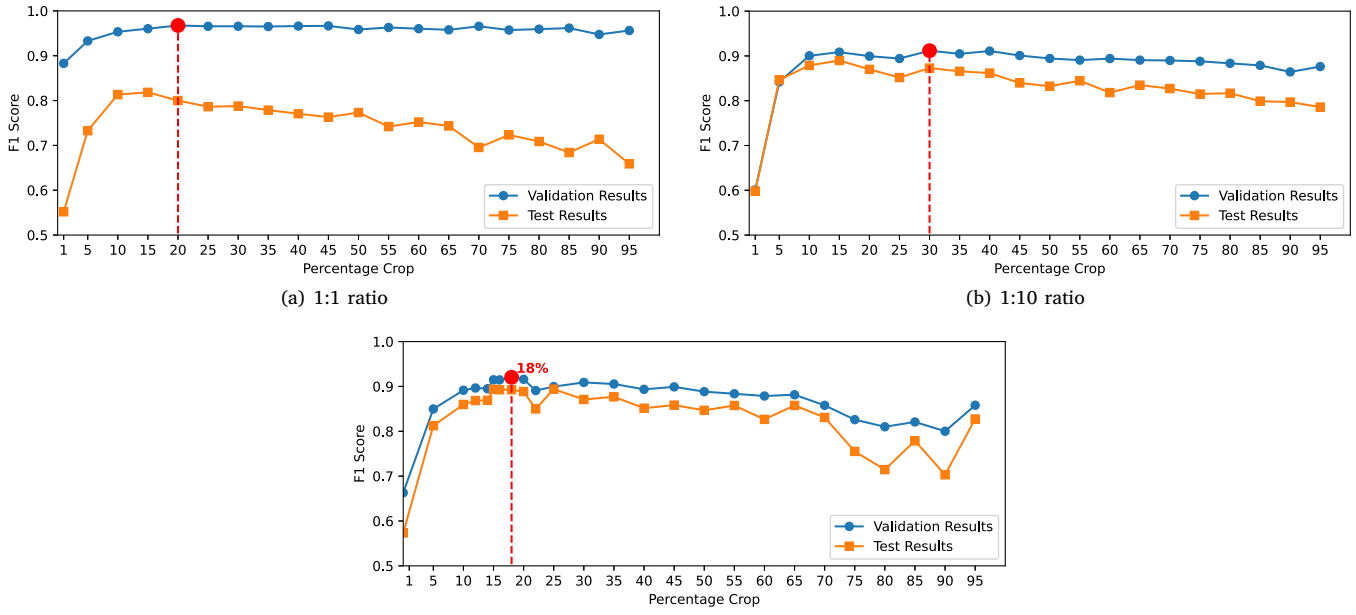
As this dataset showed promising results combined with lower computational effort, we conducted a grid search with higher resolution in the area from 10 to 24% and found the best performing *Ecklonia* patch size at 18%. The same procedure was applied to Seagrass cover with the best result at 14%, Canopy-forming Macroalgae cover at 18%, and Hardcoral cover at 20% (for extensive grid search graphs see Additional Information Appendix B.1). All subsequent experiments were initiated using the best patch size determined by this method.

3.2. Architecture comparison

To identify the most suitable architecture for our classification problem, we tested nine standard computer vision models, all pretrained on the ImageNet dataset (Kornblith et al., 2019), using a learning rate of 1e-5 and the Adam optimiser. The Inception_v3 model performed best, followed by Convnext_large, which was 3.5% lower in validation accuracy and 4.6% lower in F1-score, a trend that is even more pronounced in the campaign test results (see Table 1).

3.3. Optimiser selection

After determining the patch size and model architecture, we focused on selecting an appropriate optimiser and learning rate. We selected a set of five common optimisers in computer vision tasks (Hassan et al., 2022): AdaMax, SGD, Root Mean Square Propagation, AdamW, and Adam; and determined the most suitable learning rate using the automated learning rate finder implemented by PyTorch (Smith, 2017). As recommended by the PyTorch documentation, we tested the recommendation made by the automated learning rate finder empirically and found it to be accurate. Using a grid search, we found that the combination of the AdamW optimiser in combination with an initial learning rate of 1e-5 yielded the best results. The results can be seen in Table 2.



(a) 1:1 ratio

(b) 1:10 ratio

(c) Neighbour-distribution

Fig. 5. Displays the F1 score of the Ecklonia classifier trained on the 1:1 ratio (a), 1:10 ratio (b) and Neighbour (c) in validation and test results. The red dot with a dashed line marks the maximum validation value. Test results are averaged over eight test sites.

Table 2
Performance of optimisers trained on the *Ecklonia* dataset, with test results averaged across all eight campaign test sets, reveals similar performance in validation across all optimisers. However, AdamW exhibits slightly superior validation and campaign test results.

	Validation %		Test %	
	Acc.	F1-Score	Acc. avg.	F1-score avg.
AdamW	**94.2**	**92.3**	**96.5**	**90.8**
Adam	93.8	92.0	95.1	88.1
Adamax	93.8	91.9	95.3	88.6
SGD	93.6	91.8	95.3	88.5
RMSprop	93.2	91.0	96.2	90.2

Table 3
List of evaluated data manipulations. Bold entries indicate the combination used in the final models.

Manipulations		
RandomAffine	RandomPerspective	RandomAutocontrast
RandomEqualize	**RandomVerticalFlip**	ElasticTransform
RandomHorizontalFlip	**RandomRotation**	RandomErasing
RandomInvert	ColorJitter	RandomGrayscale

3.4. Data augmentation selection

Finally, we evaluated the benefit of data augmentation. We restricted the empirical study to basic image manipulations (Shorten and Khoshgoftaar, 2019). Using Optuna, we experimented with 12 manipulations (Table 3) in different combinations and determined that the combination of RandomHorizontalFlip, RandomVerticalFlip, RandomInvert and RandomRotate achieved the best results. This decreased the validation results slightly but generated a more stable training behaviour as well as slower convergence of the training data (see the comparison before and after in Fig. 6).

Table 4
Crossvalidation results for all classes. The score and standard deviation are calculated using 5-fold cross-validation. For extensive information on all classes see Additional Information Tables B.6–B.9.

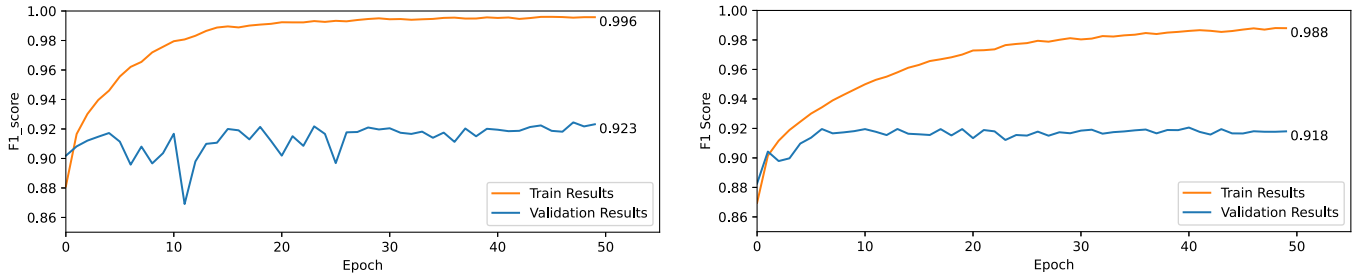
Label	F1 Valid %	F1 Test %	Acc. Valid %	Acc. Test %
Ecklonia	91.7 ±0.2	90.6 ±0.3	93.9 ±0.1	96.5 ±0.2
Seagrass	80.1 ±2.1	78.6 ±1.1	88.6 ±0.9	92.9 ±0.7
Macroalgae	92.9 ±0.2	83.2 ±0.4	92.9 ±0.2	85.9 ±0.5
Hardcoral	86.1 ±0.3	78.5 ±0.1	88.9 ±0.1	85.7 ±0.7

3.5. Cross-validation results

To summarise, based on the results of the preceding tests, we adopted transfer learning with fine-tuning of the Inception_V3 CNN model. Regarding dataset composition, the neighbour distribution performed best while requiring four times fewer computational resources. The most suitable crop sizes were 14% for Seagrass, 18% for Canopy-Forming Macroalgae, 18% for *Ecklonia*, and 20% for Hard Coral cover. For training optimisation, we used the AdamW optimiser with an initial learning rate of 1e-5. Finally, we identified the combination of RandomHorizontalFlip, RandomVerticalFlip, RandomInvert, and RandomRotate as the most effective image augmentation strategies.

To estimate performance and its variability, we applied a 5-fold split to the validation dataset and calculated the standard deviation of validation results across these splits, as shown in the Validation columns of Table 4. Each model was then evaluated on the campaign test sets, with the variation in performance across these sets reported as the standard deviation in the Test columns of Table 4.

All models demonstrated consistent performance with minimal variation. This consistency suggests strong generalisation capabilities, reducing the likelihood that favourable results arose from data misdistribution. Notably, the Seagrass model is a slight exception, displaying a validation deviation of ±2.1% for the F1-score and ±0.9% for accuracy. These variances are likely attributed to the limited training data available for Seagrass.



(a) AdamW without data augmentation. (b) AdamW with data augmentation.

Fig. 6. Performance of the *Ecklonia* model trained with the AdamW optimiser (a) without data augmentation and (b) with data augmentation. Validation results are shown in blue and training results in orange.

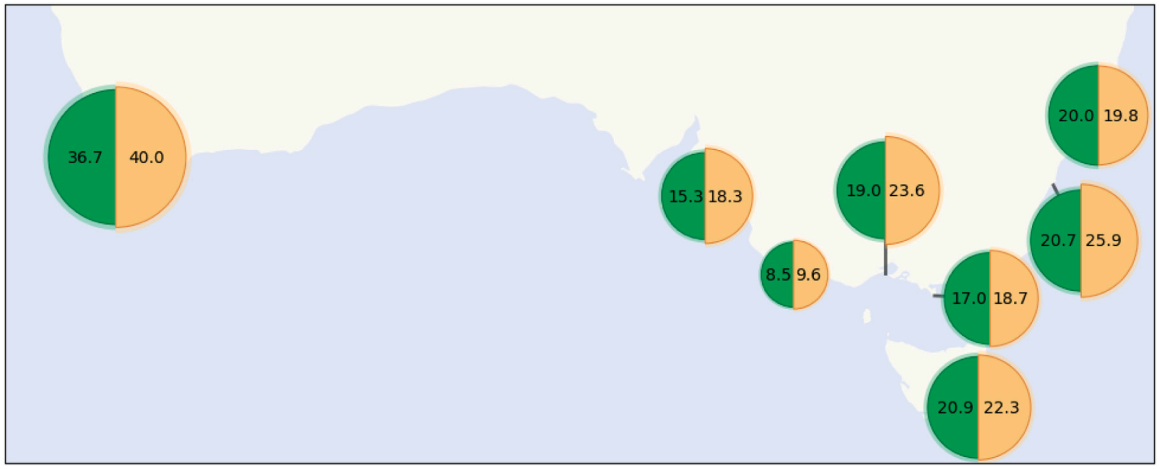


Fig. 7. Geographic comparison of annotation coverage across study sites around Southern Australia. For each deployment, half-circle symbols indicate the proportion of annotated *Ecklonia* observations in the original dataset (green left half) and after ML-assisted annotation (orange right half).

3.6. Spatial coverage analysis on unseen test sites

Applying the *Ecklonia* classifier introduced in this study, we conduct a spatial coverage analysis across eight independent test sites along the cost of Southern Australia that were not included during model training. Human-derived ground-truth *Ecklonia* annotation coverage (green) is compared with coverage obtained through ML-assisted annotation (orange). Across sites, the resulting coverage estimates exhibit broadly similar spatial patterns. The distribution further suggests a tendency of the model to slightly overclassify (see Fig. 7).

4. Discussion

This study aimed to develop and evaluate flexible, ML classifiers for benthic annotation, focusing on leveraging large, diverse datasets and assessing their transferability to campaign survey data. The classifiers performed well on both randomly selected validation datasets and independent campaign test datasets, with validation accuracy ranging from 88% for Hard Coral and Seagrass to 93% for Canopy-Forming Macroalgae and 94% for *Ecklonia*.

A central methodological contribution of this work is the neighbour-distribution approach, which improved classification performance while reducing computational cost. As with any dataset composition strategy, this approach may introduce biases that require careful evaluation, and future work could explore its extension to other annotation modalities, such as polygon-based outlines.

We further introduce an adaptive crop size which we deem more suitable for datasets containing images of highly variable resolution. Optimal patch sizes ranged from 14% for Seagrass to 20% for Hard Coral cover. We assume that smaller crops were sufficient for Seagrass due to its limited spatial extent, whereas *Ecklonia*, Canopy-Forming Macroalgae, and Hard Coral benefited from larger crops capturing more extensive structures. As image collections become increasingly heterogeneous, adaptive cropping may represent a useful standard practice.

Experiments evaluating different optimiser choices and data augmentation strategies showed no substantial improvements in classification accuracy. Performance differences between optimisers were marginal, and while data augmentation did not increase accuracy, it improved training stability and convergence behaviour, which supported more reliable comparison across experimental runs. These findings align with prior work indicating that data augmentation does not necessarily lead to accuracy gains and can, in some cases, degrade performance (Tan et al., 2022). Consequently, we do not recommend extensive optimiser or augmentation tuning as a primary means of improving performance in this setting.

In addition to standard validation and cross-validation splits, we evaluated model performance on campaign-level test sets. This analysis proved informative during model development and result interpretation, as it helped us to determine biases in the dataset composition. Further, the performance on full transects directly informs downstream applications such as the spatial coverage study, which represents a primary use case of the proposed models and demonstrates how the presented approach translates from a research setting into a practically useful tool.

As such the spatial coverage study demonstrates how the *Ecklonia* classifier can be used to extend spatial coverage across unseen sites with similar results to the human-derived annotations. The observed

tendency towards overclassification, could potentially be mitigated by filtering low-confidence predictions. However, it highlights the need for human oversight, reinforcing the role of ML as a decision-support tool within annotation workflows.

To compare each classifier presented in this study with previously published work is challenging, as the models were not trained or evaluated on the same datasets. The following comparison should therefore be interpreted as a qualitative assessment, intended to provide context rather than a strict benchmark.

Ecklonia. Compared to a recent study by Mahmood et al. (2020) focusing on *Ecklonia* classification with a dataset of comparable size and diversity, our classifier showed improvements of 10.8% in accuracy (83.4% to 94.2%) and 12.3% in F1-score (80.0% to 92.3%). In the same publication, a model was trained to annotate Canopy-Forming Macroalgae; here, our approach improved accuracy by 2.9% (90.0% to 92.9%) and F1-score by 14% (79.0% to 93.0%).

Macroalgae. To our knowledge, only one publication by D'Archino et al. (2021) has reported results exceeding ours for Canopy-Forming Macroalgae, achieving an accuracy of 97.2% (4.3% higher) and an F1-score of 98.5% (5.5% higher). However, their results for *Ecklonia* were lower, with an accuracy of 80.08% (a decrease of 14%) and an F1-score of 92.8% (similar to our results). Further, it is worth noting that their model was designed for a different task, focusing on full-frame annotation (one single annotation for each image).

Seagrass. In comparison to the publication on automated Seagrass annotation by Langlois et al. (2023), our model exhibited lower performance. Our model achieved an accuracy of 88.8%, while their model #1, which is conceptually similar to ours as a binary classifier, achieved 97.0% accuracy. It is important to note that comparing these results can be challenging due to differences in dataset handling. The authors used a dataset of similar size (5782 annotations similar to our 6800 annotations) but applied a stricter selection process, removing instances with poor visibility and other disturbances. This has shown improved model performance in other publications as well Jackett et al. (2023). However, it introduces an additional step of data filtering that is impractical for our application. as it would entail a considerable amount of manual workload, compromising the flexibility and usability of the proposed training pipeline.

Hardcoral. The hard coral classifier introduced in this publication achieved an accuracy of 88.9%, although contextualising this result within current research is difficult. The coral research community has made significant progress in addressing these issues, driven by the widespread interest in this EOV and the data-sharing platform CoralNet (Chen et al., 2021). Consequently, research in this field has evolved from binary classification (e.g., coral or not-coral) to species classification and more elaborate training and model architectures. A recent publication by Mahmood et al. (2019) with an application of binary classification achieved an overall accuracy of 96.6% (outperforming our results by 7.7%).

In summary, the *Ecklonia* binary classifier achieved the strongest performance among the evaluated models and exceeds results previously reported in the literature. This outcome is consistent with two key factors discussed above: the availability of a large and diverse training set resulting from extensive coastal mapping of *Ecklonia*, and the reduced visual variability inherent to species-level classification. In contrast, the lower performance observed for the Hard Coral classifier is plausibly explained by the high intra-class variability in morphology, colour, and texture across coral taxa. A similar effect likely contributes to the stronger performance of the Canopy-Forming Macroalgae classifier relative to Hard Coral, as it represents a more narrowly defined visual class. Taken together, these results indicate that classifiers targeting visually coherent categories tend to outperform higher-level classifiers, even when the latter benefit from larger training datasets. This pattern should be interpreted as an emerging trend rather than a definitive conclusion and warrants further investigation. For Seagrass, performance limitations are more likely attributable to the smaller dataset size than to visual variability, underscoring the continued importance of extensive training data for developing stable and reliable classifiers.

To facilitate adoption, the models presented in this study are available as a plugin within Squidle+, requiring no programming expertise as displayed in Fig. 8. Users upload their datasets to Squidle+ and define a point-based annotation scheme, after which they may select one or more task-specific binary classifiers corresponding to classes they expect to encounter (e.g., Seagrass, Macroalgae, or *Ecklonia*). Each classifier independently provides label suggestions for annotated points. When multiple classifiers assign conflicting labels for the same point those cases are explicitly highlighted in the interface and deferred to the human annotator for resolution. Further, model confidence scores can serve as a pragmatic prioritisation signal, allowing users to flag low-confidence predictions for review. However, these scores reflect internal model outputs rather than reliable measures of correctness, and deep learning models are known to exhibit overconfidence, particularly on ambiguous or previously unseen data (Ovadia et al., 2019). Confidence-based filtering should therefore be understood as a workflow optimisation strategy, with human experts retaining final oversight.

We hope that these initial and easy-to-train classifiers will stimulate further discussions and the development of additional models. By developing classifiers for the ten most common classes (see Additional Information Table C.10), we could cover approximately 50% of all publicly accessible annotations in Squidle+, significantly streamlining ecological annotation workflows.

5. Conclusion

In this work, we presented a versatile pipeline that leverages the Squidle+ platform to crowdsource a heterogeneous dataset for underwater point-count annotation. By introducing a novel neighbour-distribution approach, which we believe is a first in this field, we successfully reduced the training dataset size while enhancing both performance and consistency. Our classifiers for Canopy-Forming Macroalgae and *Ecklonia* achieved performance that is comparable to, and in several cases exceeds, results reported in the literature. These comparisons should be interpreted as a qualitative assessment intended to provide context rather than as a strict benchmark, given differences in datasets, annotation strategies, and evaluation protocols. Performance for Seagrass and Hard Coral was lower, indicating clear opportunities for further refinement and dataset expansion. Validation on geographically dispersed campaign test sets further demonstrated that the proposed models generalise across regions, supporting their applicability at a national scale.

All models and the list of associated datasets have been made publicly available, allowing users to improve and benchmark their own models against our results. Further, the models have been integrated into the Squidle+ annotation interface, which allows users to apply the models to their own datasets, significantly lowering the barrier to implementation, as well as reducing the need for advanced technical knowledge. This streamlined annotation process frees researchers to concentrate on more complex and nuanced examples, ensuring their expertise is directed towards the most challenging aspects of analysis.

6. Data sharing

All algorithms used for dataset creation, model training, and deployment in this study are publicly available via the following GitHub repository: https://github.com/leoguen/Squidle_ML_Algorithm/.

An extensive overview of both publicly available and non-public datasets used in this study is provided in Appendix C.3. While some datasets are currently classified as non-public, data owners may be

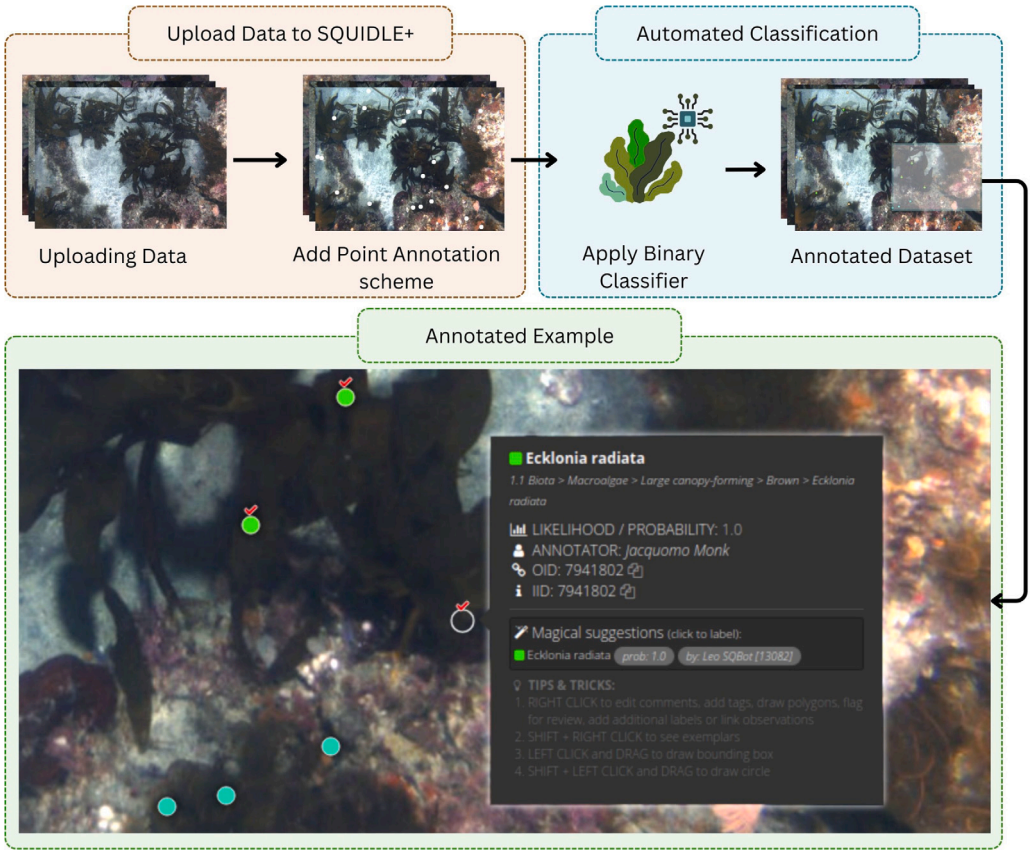


Fig. 8. Implementation of the *Ecklonia* classifier in Squidle+. The colour of the dots indicate human made annotations (e.g. "green" represents *Ecklonia*), while ticks denote annotations correctly suggested by the classifier. The pop-up window shows model confidence scores ("Magical suggestions").

willing to grant access for the purpose of model training upon request. In addition, several datasets presently classified as non-public may become publicly accessible in the future as contributing researchers complete their analyses, and we are actively in contact with multiple groups to facilitate the release of these data.

CRediT authorship contribution statement

Leonard Günzel: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Jacquomo Monk:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Chris Jackett:** Writing – review & editing, Supervision, Software, Methodology. **Ariell Friedman:** Writing – review & editing, Software, Methodology, Data curation, Conceptualization. **Ashlee Bastiaansen:** Writing – review & editing, Methodology, Data curation. **Ardalan Najafi:** Writing – review & editing, Supervision. **Alberto García Ortiz:** Writing – review & editing, Supervision. **Neville Barrett:** Writing – review & editing, Supervision.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT in order to improve grammar and readability. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank the Integrated Marine Observing System (IMOS) for supporting the AUV monitoring program, ReefLifeSurvey and Understanding of Marine Imagery (UMI), which contributes to the annotation repository (Squidle+) used in this work. IMOS is enabled by the National Collaborative Research Infrastructure Strategy (NCRIS). The Australian Centre for Field Robotics at the University of Sydney is also thanked for operating the AUVs. Agencies contributing to UMI are also thanked for their support. In particular, Elizabeth Oh, for providing additional training data associated with ReefLifeSurvey Photoquadrats. We gratefully acknowledge Parks Australia for supporting the collection and annotation of imagery contributing to this work. This work has been carried out as part of the Safeguard project. The Norwegian Research Council, Norway and the Petromaks 2 programme are acknowledged as the main sponsors of project number 344166.

Appendix A. Additional background information

A.1. ML models

The models used in this work are deep neural networks, which employ hierarchical architectures to learn high-level abstractions from the data. Each layer in the model captures different patterns, with some layers focusing on simpler details like shades and edges, while others build on this information to analyse more complex features such as patterns. One of the most successful types of deep neural networks is the Convolutional Neural Network (CNN), which derives its name from the mathematical operation of convolution between matrices. The field has further advanced from this development by building upon

the introduction of transformers (Vaswani et al., 2017) (a different ML architecture) and adapting those principles into the realm of computer vision (Dosovitskiy et al., 2020), creating the field of vision transformers. Vision transformers have demonstrated an improvement over CNNs in classical ML benchmarks (Paul and Chen, 2022). However, it is worth noting that CNNs remain widely accepted as suitable architectures for various computer vision tasks due to their relatively high accuracy and training efficiency (Guo et al., 2016). Given the well-established position of CNNs in the field and the robust infrastructure available for further development and improvement, we have chosen to leverage this architecture for our work.

CNNs are made up of a combination of three types of layers: convolutional layers, pooling layers, and fully connected layers (represented in Fig. 4). Generally speaking convolutional layers in combination with pooling layers are used to extract features and the combination of those extracted features is then passed to a number of fully connected layers for classification. For a more in-depth discussion see Albawi et al. (2017). In the following, the methods will be explained using *Ecklonia* as an example, but have been and can be applied similarly to other species of interest.

CNN models are iteratively trained through a series of epochs. In one epoch the model is given each image from the training dataset as input, and the training takes place in two phases: a forward stage and a backward stage (backpropagation). In the forward stage, the model predicts on an input image using the current weights. Whenever the model predicts the label of an image, it gives a probability for each label that it is trained on (in this case two). The label with the higher probability will be applied to the image. This prediction output is then compared to the ground truth label, to calculate the loss. In the second stage, the backward phase, the loss is then used to calculate the gradients and modify each parameter and thus adapt the model; the model is "learning". Once all training images have been processed, the trained model predicts on the validation set. For this part of the epoch the backward stage is deactivated, thus the model cannot learn. This guarantees that the model cannot adapt to the validation dataset, it serves as a control set. A good model, while training should increase the performance equally in training and validation results. Thus if we plot both results per epoch of training and validation, the model should converge and both values should approximate each other until reaching a stable phase.

A.2. Evaluation parameters

We have to introduce a few more parameters to gain a better understanding of our model's behaviour. Whenever our model makes a prediction, we compare that prediction to the ground truth and give it one of the following labels:

- True Positive: The image depicts *Ecklonia* and the model predicts "*Ecklonia*"
- True Negative: The image depicts sand and the model predicts "Others"
- False Positive: The image depicts sand but the model predicts "*Ecklonia*"
- False Negative: The image depicts *Ecklonia* but the model predicts "Others"

If we accumulate those labels over a set of predictions, we can calculate several parameters to evaluate the model's performance on this prediction set. A common measure in the field is Accuracy. To calculate the Accuracy, we look at the total number of correct predictions (True Positives and True Negatives), overall predictions:

$$Accuracy = \frac{TruePositives + TrueNegatives}{TruePositives + TrueNegatives + FalsePositives + FalseNegatives} \tag{A.1}$$

Table B.5
Crop percentage grid search results.

Label name	Crop percentage	F1-score
Ecklonia	18	0.908
Seagrass	14	0.802
Macroalgae	18	0.924
Hardcoral	20	0.863

Accuracy is a good measure to describe a model's behaviour if the target and non-target classes are equally distributed, as in our 1:1 dataset. This is not the case for the 1:10 dataset. Here, our target class makes up approximately 9% of the dataset. If a model were to just define everything as 'Others' and not label *Ecklonia* at all. This model would still end up with a 91% Accuracy, which can be considered a reasonable result for our problem:

$$Accuracy = \frac{0 + 620,000}{0 + 620,000 + 60,000 + 0} = 0.9117 \tag{A.2}$$

Therefore, we have to introduce other evaluation parameters Precision, Recall, and F1-score. In the case of *Ecklonia* as the target, Precision would define the amount of correctly identified *Ecklonia* entries, divided by all *Ecklonia* predictions, including the incorrect ones. Whereas Recall defines the amount of predicted *Ecklonia* entries over the actual amount of *Ecklonia* entries:

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{A.3}$$

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{A.4}$$

These two metrics can be combined to form the F1-Score, the harmonic mean of Precision and Recall. The F1-score combines these two metrics into a single score that ranges from 0 to 1, with 1 being the best possible score:

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{A.5}$$

The F1-score is a widely used performance metric for classification tasks and is often preferred over accuracy when dealing with imbalanced datasets. In this analysis, we will therefore focus primarily on the F1-score. Nevertheless, accuracy remains a common and well-recognised evaluation metric in the field. For completeness, we will also report accuracy in the final results.

Appendix B. Extended results

B.1. Grid search for EOVs

Fig. B.9 summarises the influence of crop size on model performance when using the neighbour condition with a 1:1 sampling ratio. Accuracy and F1-score are shown as functions of crop percentage for all four target classes, with results reported separately for validation and campaign-level test data. Across classes, the discrepancy between validation and test performance is reduced compared to the 1:1 baseline configuration (Fig. 5(a)), indicating improved generalisation. Performance peaks consistently within a crop percentage range of approximately 10%–25%, motivating a higher-resolution search in this region using 2% crop increments to identify the optimal crop size summarized in Table Table B.5.

B.2. Crossvalidation results

To assess the robustness and stability of the trained models, we performed 5-fold cross-validation for each target class: Seagrass, Canopy-Forming Macroalgae, Hard Coral, and *Ecklonia*. In this procedure, the

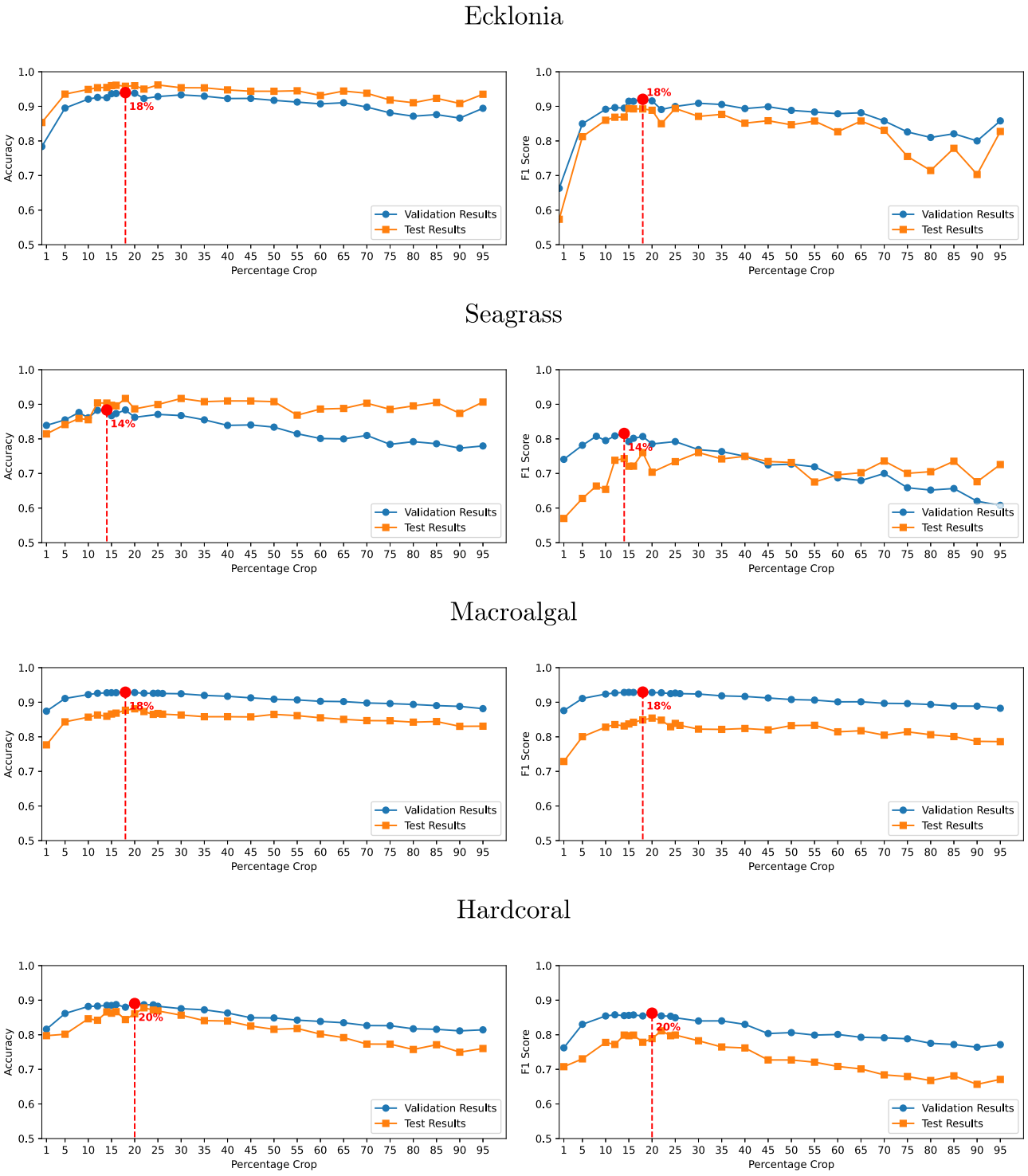


Fig. B.9. Accuracy (left) and F1-score (right) as functions of crop percentage for models trained using the neighbour condition with a 1:1 ratio. Rows correspond to *Ecklonia*, Seagrass, Macroalgae, and Hard coral. Validation results are shown in blue (circles) and test results in orange (squares); the red marker indicates the maximum validation score. Test results are averaged across eight sites for *Ecklonia* and three sites for the remaining classes.

Table B.6
Crossvalidation Seagrass.

	f1_valid	*acc_valid*	*f1_test*	*f1_test_avg*	*acc_test*	*acc_test_avg*
1	0.77872	0.87811	[0.80478, 0.74654, 0.80093]	0.78408	[0.95889, 0.88629, 0.94267]	0.92928
2	0.80840	0.88655	[0.78891, 0.75036, 0.75836]	0.76588	[0.93400, 0.88215, 0.94547]	0.92054
3	0.79941	0.87988	[0.79826, 0.79087, 0.74859]	0.77924	[0.93800, 0.95386, 0.87733]	0.92306
4	0.82830	0.90031	[0.78298, 0.76090, 0.83036]	0.79141	[0.95721, 0.89042, 0.94933]	0.93232
5	0.78803	0.88366	[0.77451, 0.84746, 0.80193]	0.80797	[0.90489, 0.96979, 0.94533]	0.94001
Avg.	0.80099	0.88590	[0.78908, 0.77984, 0.78844]	0.78552	[0.94660, 0.93610, 0.95240]	0.92944
Std. Dev.	0.02086	0.00861	[0.01242, 0.04538, 0.03315]	0.01113	[0.02449, 0.03652, 0.03094]	0.00653

Table B.7
Crossvalidation Macroalgae.

	f1_valid	*acc_valid*	*f1_test*	*f1_test_avg*	*acc_test*	*acc_test_avg*
1	0.92594	0.92681	[0.86821, 0.87518, 0.76178]	0.83505	[0.82146, 0.92069, 0.85151]	0.86455
2	0.92973	0.92948	[0.73236, 0.86981, 0.88718]	0.82978	[0.81544, 0.91431, 0.83947]	0.85641
3	0.92868	0.92876	[0.88716, 0.72962, 0.86992]	0.82890	[0.84103, 0.82466, 0.91249]	0.85940
4	0.93089	0.93099	[0.89051, 0.86980, 0.75250]	0.83760	[0.84730, 0.91158, 0.83389]	0.86426
5	0.92742	0.92568	[0.86803, 0.74155, 0.88229]	0.83062	[0.91158, 0.82047, 0.82929]	0.85378
Avg.	0.92873	0.92876	[0.86905, 0.81765, 0.85053]	0.83219	[0.86776, 0.89854, 0.87773]	0.85900
Std. Dev.	0.00214	0.00234	[0.07191, 0.07027, 0.06650]	0.00413	[0.04067, 0.04324, 0.03568]	0.00546

Table B.8
Crossvalidation Hardcoral.

	f1_valid	*acc_valid*	*f1_test*	*f1_test_avg*	*acc_test*	*acc_test_avg*
1	0.85793	0.88783	[0.86777, 0.80289, 0.73356]	0.80141	[0.84991, 0.89658, 0.86000]	0.86883
2	0.85902	0.88822	[0.85085, 0.72318, 0.74306]	0.77236	[0.83490, 0.85455, 0.85958]	0.84967
3	0.86070	0.88937	[0.72362, 0.85232, 0.77634]	0.78409	[0.85000, 0.83583, 0.88520]	0.85701
4	0.86218	0.88757	[0.87653, 0.72250, 0.77108]	0.79004	[0.85835, 0.84636, 0.87381]	0.85951
5	0.86385	0.89075	[0.74957, 0.86246, 0.71405]	0.77536	[0.86243, 0.84053, 0.83909]	0.84735
Avg.	0.86054	0.88855	[0.81375, 0.79267, 0.75142]	0.78465	[0.85132, 0.87467, 0.86314]	0.85687
Std. Dev.	0.00260	0.00142	[0.05672, 0.06269, 0.04659]	0.01248	[0.01076, 0.02122, 0.01172]	0.00784

Table B.9
Crossvalidation Ecklonia.

	f1_valid	*acc_valid*	*f1_test*	*f1_test_avg*	*acc_test*	*acc_test_avg*
1	0.91898	0.93922	[0.77205, 0.93258, 0.95385, 0.88844 0.89474, 0.89559, 0.93804, 0.97935]	0.90683	[0.90558, 0.97601, 0.96521, 0.94986, 0.96554, 0.98182, 0.97400, 0.99160]	0.96370
2	0.91984	0.94015	0.89162, 0.95183, 0.90868, 0.92256, 0.76667, 0.95735, 0.88729, 0.97527]	0.90766	[0.95169, 0.96361, 0.98384, 0.97241, 0.90351, 0.98200, 0.96319, 0.99000]	0.96378
3	0.92093	0.94058	0.95275, 0.98410, 0.86836, 0.74839, 0.87649, 0.94959, 0.86621, 0.95360]	0.89994	[0.98040, 0.99360, 0.95536, 0.89249, 0.94348, 0.98281, 0.97616, 0.96521]	0.96119
4	0.91802	0.93904	0.87411, 0.89524, 0.95494, 0.97738, 0.89855, 0.78912, 0.91319, 0.95569]	0.90728	[0.95850, 0.98222, 0.98120, 0.99080, 0.95533, 0.91454, 0.96921, 0.96681]	0.96483
5	0.91421	0.93716	0.94768, 0.77496, 0.97872, 0.90909, 0.96372, 0.89362, 0.88732, 0.93810]	0.91165	[0.97880, 0.90834, 0.99160, 0.97103, 0.97321, 0.95442, 0.98061, 0.97921]	0.96715
Avg.	0.91660	0.93923	0.86722, 0.90733, 0.93352, 0.88888, 0.88143, 0.91439, 0.91859, 0.95816	0.90617	0.95307, 0.96465, 0.97520, 0.95032, 0.93891, 0.96271, 0.96063, 0.97438	0.96471
Std. Dev.	0.00219	0.00129	0.07919, 0.09250, 0.10924, 0.07112, 0.07312, 0.05858, 0.03119, 0.01427	0.00348	0.02418, 0.03017, 0.01531, 0.01848, 0.02798, 0.01336, 0.00613, 0.01013	0.00230

Table C.10
25 most common publicly available annotations on Squidle accessed on 27th of April 2023.

Idx	Annotation	Count
1	Catami 1.4 > Coarse sand (with shell fragments)	18,436
2	RLS Catalogue > Turfing algae (<2 cm high algal/sediment mat on rock)	10,427
3	Australian Morphospecies Catalogue > Bryozoa/Cnidaria Matrix	9857
4	RLS Catalogue > Crustose coralline algae	5463
5	Australian Morphospecies Catalogue > Ecklonia radiata	4651
6	Catami 1.4 > Sand/mud (<2mm)	4617
7	Catami 1.4 > Fine sand (no shell fragments)	4310
8	RLS Catalogue > Medium foliose brown algae	3263
9	Catami 1.4 > Filamentous/filiform	3229
10	Catami 1.4 > Corals	3215
11	RLS Catalogue > Unscorable	2948
12	Australian Morphospecies Catalogue > Bryozoa/Cnidaria/Sponge Matrix	2907
13	Australian Morphospecies Catalogue > Foliose	2445
14	RLS Catalogue > Sand	1991
15	Australian Morphospecies Catalogue > Bryozoa/Cnidaria/Encrusting Sponge Matrix	1982
16	Catami 1.4 > Mud/silt (<64um)	1958
17	Catami 1.4 > Calcareous	1952
18	Australian Morphospecies Catalogue > Posidonia spp	1921
19	Australian Morphospecies Catalogue > Cup Red Smooth	1908
20	RLS Catalogue > Filamentous brown algae_epiphyte	1848
21	Catami 1.4 > Red	1748
22	Catami 1.4 > Sea urchins	1707
23	Catami 1.4 > Rock	1688
24	Australian Morphospecies Catalogue > Turf/Sediment/Silt matrix	1687
25	Australian Morphospecies Catalogue > Phyllospora comosa	1369

dataset is split into five equally sized subsets. For each fold, four subsets are used for training, while the remaining subset is held out for validation. This process is repeated five times so that each subset serves once as the validation set, reducing the influence of any single train–validation split on the reported performance.

Tables B.6–B.9 summarise the results of this cross-validation. Each row (labelled 1–5) corresponds to one cross-validation run. For each run, the tables report the F1-score and accuracy on the validation set (*f1_valid*, *acc_valid*). In addition, performance on the independent campaign-level test sets is reported. Because multiple test campaigns were evaluated, the per-campaign F1-scores and accuracies are listed in brackets, with the corresponding averaged values shown in the adjacent columns (*f1_test_avg*, *acc_test_avg*).

The final two rows in each table present the mean performance across all five folds and the corresponding standard deviation. These values provide an indication of both the overall model performance and its variability across different training splits, allowing an assessment of model stability and sensitivity to data partitioning.

Appendix C. Common annotations and data accessibility

C.1. Most common Squidle+ annotations

The following table provides the 25 most common publicly available classes on Squidle+ with their respective annotation count. The table is referred to in the Discussion.

C.2. Testset distribution

In the subsequent section, tables with details of the test sets are presented for each of the four classes. These details include the campaign name associated with each set, providing context for the data collection source. Additionally, the percentage cover is listed, giving the proportion of annotations of a specific class in relation to the overall annotation count (the last column) within that particular campaign.

C.3. Data accessibility

Below, the public and private accessibility of the datasets used for training the *Ecklonia* (Table C.13), Hardcoral (Table C.16), Macroalgal (Table C.15), and Seagrass (Table C.14) models is shown. Campaigns contributing less than 0.5% of the total annotations are omitted for clarity. It is important to note that some datasets currently classified as non-public may become publicly accessible in the future as contributing researchers complete their analyses; we are actively in contact with several of these groups to support the release of their data. Furthermore, since the datasets used in this study were compiled in April 2023, a substantial number of additional campaigns have since been added to the platform. As a result, contemporary algorithms trained exclusively on publicly available data are likely to have access to a larger volume of training data than was available for the models presented here.

C.3.1. Ecklonia

For *Ecklonia*, 97.2% of the annotations assigned to the target class (*Ecklonia*) used for model training are publicly available, compared to 89.5% of all annotations used for training and validation.

Table C.11
Distribution of Ecklonia annotations in test sets and the total amount of annotations.

Campaign name	Percentage of Ecklonia annotations in testset	No. annotations	No. images
RLS_Gulf St Vincent_2012	15.3	1277	251
RLS_Jervis Bay Marine Park_2015	20.7	1097	216
RLS_Port Phillip Heads_2010	19.0	1451	248
r20150210_010919_broughton_east_biebg_03c_broad	20.04	2500	100
DiscoveryBay202112	8.5	2500	99
Tasmania201205	20.9	2500	100
Wilsonsprom201603SS	17.0	2500	100
WA_SW_202103	36.7	2500	100

Table C.12
Distribution of Seagrass, Macroalgal canopy, and Hard coral cover annotations in test sets and the total amount of annotations.

	Campaign name	Percentage of object of interest annotations in testset	No. annotations	No. images
Seagrass **cover**	RLS_Port Phillip Heads_2010	20.74	1451	248
	RLS_Port Phillip Bay_2016	14.1	1500	296
	RLS_Port Phillip Bay_2017	10.1	1192	232
Macroalgal **canopy cover**	RLS_Gulf St Vincent_2012	67.3	1277	251
	RLS_Port Phillip Bay_2017	31.7	1192	232
	RLS_Jervis Bay Marine Park_2015	30.1	1097	216
Hard coral **cover**	RLS_Queensland (other)_2015	56.1	1066	390
	RLS_Norfolk Island_2013	23.3	1100	217
	RLS_Ningaloo Marine Park_2016	23.3	1054	189

Table C.13
Campaign-wise contribution of *Ecklonia* annotations to the dataset.

No.	Campaign	Count	%	Access	No.	Campaign	Count	%	Access
1	RLS_Port Phillip Heads_2020	5315	8.7	Public	24	RLS_Sydney_2010	602	1.0	Public
2	Tasmania201306	4752	7.8	Public	25	RLS_Sydney_2011	525	0.9	Public
3	RLS_Wilsons Promontory_2020	4385	7.2	Public	26	Tasmania201205	522	0.9	Public
4	Tasmania201610	4070	6.7	Public	27	RLS_Victor Harbor_2018	579	0.9	Public
5	Tasmania201106	3519	5.8	Public	28	PS201502	501	0.8	Public
6	Tasmania200810	2825	4.6	Public	29	RLS_Sydney_2018	405	0.7	Public
7	WA_SW_202103	2196	3.6	Public	30	RLS_Gulf St Vincent_2019	436	0.7	Private
8	Tasmania201006	1813	3.0	Public	31	Wilsonsprom201603SS	424	0.7	Private
9	RLS_Victor Harbor_2013	1449	2.4	Public	32	RLS_Batemans Marine Park_2010	436	0.7	Public
10	Tasmania201406	1406	2.3	Public	33	RLS_Adelaide_2017	417	0.7	Public
11	RLS_Rottnest Island_2021	1434	2.3	Public	34	RLS_Victor Harbor_2017	392	0.6	Public
12	Batemans201011	1287	2.1	Public	35	RLS_Port Phillip Bay_2013	372	0.6	Public
13	RLS_Adelaide_2020	1237	2.0	Public	36	RLS_Sydney_2017	383	0.6	Public
14	RLS_Adelaide_2019	1156	1.9	Public	37	RLS_Sydney_2015	353	0.6	Public
15	RLS_Port Phillip Heads_2013	1074	1.8	Public	38	RLS_Port Phillip Bay_2010	355	0.6	Public
16	RLS_Port Phillip Heads_2012	965	1.6	Public	39	RLS_Jervis Bay Marine Park_2013	365	0.6	Public
17	RLS_Gulf St Vincent_2020	964	1.6	Public	40	RLS_Adelaide_2018	378	0.6	Public
18	RLS_Victor Harbor_2020	944	1.5	Public	41	RLS_Port Phillip Bay_2020	341	0.6	Public
19	Batemans201211	933	1.5	Public	42	RLS_Encounter_2014	300	0.5	Private
20	RLS_Port Phillip Bay_2009	783	1.3	Public	43	RLS_Sydney_2014	298	0.5	Public
21	RLS_Sydney_2020	756	1.2	Public	44	RLS_Sydney_2009	290	0.5	Public
22	RLS_Sydney_2019	668	1.1	Public	45	RLS_Port Phillip Heads_2011	276	0.5	Public
23	RLS_Port Phillip Heads_2009	648	1.1	Public					

C.3.2. Seagrass

For Seagrass, 96.7% of the target-class annotations used for training are publicly available, while 85.8% of all annotations used for training and validation are public.

C.3.3. Macroalgae

For Macroalgae, 95.5% of the target-class annotations used in model training are publicly available, with 85.8% of the full training and validation annotation set being public.

C.3.4. Hardcoral

For Hardcoral, 49.7% of the target-class annotations used for training are publicly available, and 86.1% of all annotations included in training and validation are public.

Data availability

Further, most of the data used to train the algorithm is publicly available on Squidle+ (https://squidle.org/)

Table C.14
Campaign-wise contribution of Seagrass annotations to the dataset.

No.	Campaign	Count	%	Access	No.	Campaign	Count	%	Access
1	RLS_Port Phillip Heads_2020	1051	15.3	Public	19	RLS_Port Stephens_2021	75	1.1	Public
2	WA_SW_202103	1042	15.2	Public	20	RLS_Port Phillip Bay_2013	74	1.1	Public
3	RLS_Port Phillip Bay_2020	587	8.6	Public	21	RLS_Gulf St Vincent_2019	68	1.0	Private
4	RLS_Port Phillip Bay_2009	429	6.3	Public	22	RLS_Port Stephens_2017	71	1.0	Public
5	RLS_Port Phillip Heads_2013	368	5.4	Public	23	RLS_Port Phillip Heads_2009	65	0.9	Public
6	RLS_Adelaide_2020	354	5.2	Public	24	RLS_Port Stephens_2010	64	0.9	Public
7	RLS_Port Phillip Heads_2012	275	4.0	Public	25	RLS_Sydney_2019	58	0.8	Public
8	RLS_Port Phillip Bay_2019	252	3.7	Public	26	RLS_Adelaide_2016	46	0.7	Public
9	RLS_Port Phillip Heads_2011	228	3.3	Public	27	RLS_Jervis Bay Marine Park_2013	45	0.7	Public
10	RLS_Port Phillip Bay_2010	150	2.2	Public	28	RLS_Adelaide_2015	41	0.6	Public
11	RLS_Port Stephens_2009	129	1.9	Public	29	RLS_Port Phillip Bay_2011	41	0.6	Public
12	RLS_Yorke Peninsula_2022	108	1.6	Private	30	RLS_Jervis Bay Marine Park_2010	33	0.5	Public
13	RLS_Melilla Coast_2015	111	1.6	Public	31	RLS_Jervis Bay Marine Park_2015	32	0.5	Public
14	RLS_Port Stephens_2018	108	1.6	Public	32	WA201104	37	0.5	Public
15	RLS_Port Phillip Bay_2018	107	1.6	Public	33	RLS_New South Wales (Other)_2022	37	0.5	Public
16	RLS_Gulf St Vincent_2020	103	1.5	Public	34	RLS_Port Stephens_2019	37	0.5	Public
17	RLS_Port Stephens_2013	101	1.5	Public	35	RLS_Port Stephens_2016	35	0.5	Public
18	RLS_Rottnest Island_2021	74	1.1	Public					

Table C.15
Campaign-wise contribution of Macroalgae annotations to the dataset.

No.	Campaign	Count	%	Access	No.	Campaign	Count	%	Access
1	WA201104	52 488	25.9	Public	19	WA_SW_202103	2215	1.1	Public
2	RLS_Wilsons Promontory_2020	16 544	8.2	Public	20	RLS_Adelaide_2020	2113	1.0	Public
3	RLS_Port Phillip Heads_2020	9157	4.5	Public	21	RLS_Port Phillip Heads_2012	2067	1.0	Public
4	Tasmania201610	8842	4.4	Public	22	SEQueensland201010	1874	0.9	Public
5	Tasmania201306	7204	3.6	Public	23	RLS_Rottnest Island_2021	1808	0.9	Public
6	WA201204	6185	3.1	Public	24	RLS_Cape Howe_2011	1713	0.8	Public
7	RLS_Gulf St Vincent_2020	5982	3.0	Public	25	RLS_Victor Harbor_2018	1595	0.8	Public
8	Tasmania201106	5800	2.9	Public	26	RLS_Gulf St Vincent_2017	1576	0.8	Public
9	Tasmania200810	5294	2.6	Public	27	RLS_Port Phillip Bay_2013	1484	0.7	Public
10	RLS_Gulf St Vincent_2019	4759	2.4	Private	28	RLS_Port Phillip Heads_2009	1224	0.6	Public
11	Tasmania201006	4014	2.0	Public	29	RLS_Encounter_2014	1281	0.6	Private
12	Tasmania201406	3279	1.6	Public	30	RLS_Batemans Marine Park_2010	1201	0.6	Public
13	RLS_Victor Harbor_2013	2718	1.3	Public	31	RLS_Port Phillip Bay_2010	1115	0.6	Public
14	RLS_Cape Howe_2021	2613	1.3	Public	32	Batemans201011	1290	0.6	Public
15	RLS_Adelaide_2019	2724	1.3	Public	33	RLS_Port Phillip Heads_2011	1039	0.5	Public
16	RLS_Port Phillip Bay_2009	2611	1.3	Public	34	Batemans201211	934	0.5	Public
17	RLS_Port Phillip Heads_2013	2444	1.2	Public	35	RLS_Victoria (other)_2017	961	0.5	Public
18	RLS_Victor Harbor_2020	2349	1.2	Public	36	RLS_Victor Harbor_2017	1109	0.5	Public

Table C.16
Campaign-wise contribution of Hardcoral annotations to the dataset.

No.	Campaign	Count	%	Access	No.	Campaign	Count	%	Access
1	WA201104	10,841	12.9	Public	25	RLS_Lord Howe Island_2014	852	1.0	Private
2	WA201304	8811	10.4	Public	26	RLS_Lord Howe Island_2018	774	0.9	Private
3	EMR202001	5421	6.4	Public	27	RLS_Elizabeth and Middleton Reefs_2018	781	0.9	Public
4	RLS_North West Shelf_2013	4761	5.6	Private	28	RLS_Northern Territory (Other)_2015	752	0.9	Private
5	RLS_North West Shelf_2018	4054	4.8	Public	29	RLS_Norfolk Island_2021	787	0.9	Private
6	RLS_Great Barrier Reef_2020	2248	2.7	Private	30	RLS_Lord Howe Island_2012	677	0.8	Private
7	RLS_Coral Sea_2013	2168	2.6	Private	31	RLS_Great Barrier Reef_2017	649	0.8	Private
8	RLS_Coral Sea_2017	2103	2.5	Private	32	RLS_Pulau Jamdena_2015	660	0.8	Public
9	RLS_Coral Sea_2016	1909	2.3	Private	33	RLS_Elizabeth and Middleton Reefs_2013	678	0.8	Private
10	RLS_Coral Sea_2020	1796	2.1	Private	34	RLS_Coral Sea_2022	663	0.8	Private
11	RLS_Great Barrier Reef_2010	1627	1.9	Private	35	RLS_Lord Howe Island_2016	710	0.8	Private
12	RLS_Capricorn Group_2020	1500	1.8	Private	36	RLS_Norfolk Island_2009	628	0.7	Private
13	RLS_Coral Sea_2015	1549	1.8	Private	37	RLS_Lord Howe Island_2020	602	0.7	Private
14	RLS_Great Barrier Reef_2016	1383	1.6	Private	38	RLS_Coral Sea_2021	517	0.6	Private
15	RLS_Ningaloo Marine Park_2019	1275	1.5	Public	39	RLS_Ningaloo Marine Park_2010	500	0.6	Private
16	RLS_Great Barrier Reef_2015	1184	1.4	Private	40	RLS_Solitary Islands_2008	475	0.6	Public
17	RLS_Capricorn Group_2015	1067	1.3	Private	41	RLS_Abrolhos Islands_2008	527	0.6	Private
18	CoralSea202008	1005	1.2	Private	42	RLS_Abrolhos Islands_2021	495	0.6	Private
19	RLS_Ningaloo Marine Park_2012	929	1.1	Public	43	RLS_Coral Sea_2012	397	0.5	Private
20	RLS_Ashmore/Cartier_2021	938	1.1	Private	44	RLS_Coral Sea_2018	424	0.5	Private
21	RLS_Capricorn Group_2017	917	1.1	Private	45	RLS_Abrolhos Islands_2013	429	0.5	Private
22	RLS_Ningaloo Marine Park_2017	874	1.0	Public	46	RLS_Oceanic Shoals_2021	401	0.5	Private
23	RLS_Ningaloo Marine Park_2015	816	1.0	Public	47	RLS_Pulau Naira_2015	456	0.5	Public
24	RLS_Lord Howe Island_2010	809	1.0	Private	48	RLS_Carpentaria_2021	457	0.5	Private

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org, URL https://www.tensorflow.org/.

Aguzzi, J., Costa, C., Robert, K., Matabos, M., Antonucci, F., Juniper, S.K., Menesatti, P., 2011. Automated image analysis for the detection of benthic crustaceans and bacterial mat coverage using the VENUS undersea cabled network. Sensors 11 (11), 10534–10556. http://dx.doi.org/10.3390/s111110534, URL https://www.mdpi.com/1424-8220/11/11/10534.

Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M., 2019. Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. http://dx.doi.org/10.1145/3292500.3330701.

Albawi, S., Mohammed, T.A., Al-Zawi, S., 2017. Understanding of a convolutional neural network. In: 2017 International Conference on Engineering and Technology. ICET, pp. 1–6. http://dx.doi.org/10.1109/ICEngTechnol.2017.8308186.

Beijbom, O., Treibitz, T., Kline, D.I., Eyal, G., Khen, A., Neal, B., Loya, Y., Mitchell, B.G., Kriegman, D., 2016. Improving automated annotation of benthic survey images using wide-band fluorescence. Sci. Rep. 6 (1), http://dx.doi.org/10.1038/srep23166.

Bewley, M., Douillard, B., Nourani-Vatani, N., Friedman, A., Pizarro, O., Williams, S., 2012. Automated species detection: An experimental approach to kelp detection from sea-floor AUV images. In: Proc Australas Conf Rob Autom, vol. 2012.

Chen, Q., Beijbom, O., Chan, S., Bouwmeester, J., Kriegman, D., 2021. A new deep learning engine for CoralNet. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops. ICCVW, http://dx.doi.org/10.1109/iccvw54120.2021.00412.

Choi, D., Shallue, C.J., Nado, Z., Lee, J., Maddison, C.J., Dahl, G.E., 2019. On empirical comparisons of optimizers for deep learning. arXiv:1910.05446.

Chu, B., Madhavan, V., Beijbom, O., Hoffman, J., Darrell, T., 2016. Best practices for fine-tuning visual classifiers to new domains. In: Hua, G., Jégou, H. (Eds.), Computer Vision – ECCV 2016 Workshops. Springer International Publishing, Cham, pp. 435–442.

D'Archino, R., Schimel, A.C., Peat, C., Anderson, T.J., 2021. Automated Detection of Large Brown Macroalgae Using Machine Learning Algorithms: A Case Study from Island Bay, Wellington. Ministry for Primary Industries.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. ImageNet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255. http://dx.doi.org/10.1109/CVPR.2009.5206848.

Denuelle, A., Dunbabin, M., 2010. Kelp detection in highly dynamic environments using texture recognition. In: The Australasian Conference on Robotics & Automation. Citeseer.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. CoRR abs/2010.11929, arXiv:2010.11929, URL https://arxiv.org/abs/2010.11929.

Durden, J.M., Schoening, T., Althaus, F., Friedman, A., Garcia, R., Glover, A.G., Greinert, J., Stout, N.J., Jones, D., Jordt, A., et al., 2016. Perspectives in visual imaging for marine biology and ecology: From acquisition to understanding. Ocean. Mar. Biology - Annu. Rev. 1–73. http://dx.doi.org/10.1201/9781315368597-2.

Erhan, D., Courville, A., Bengio, Y., Vincent, P., 2010. Why does unsupervised pre-training help deep learning? In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. JMLR Workshop and Conference Proceedings, pp. 201–208.

Friedman, A., Monk, J., Pizarro, O., Lindsay, D.J., Oh, E., Thornton, B., Carroll, A.G., Przeslawski, R., Williams, S.B., 2026. Squidle+: a collaborative platform to manage, discover and annotate marine imagery. Front. Mar. Sci. 12, 1677103. http://dx.doi.org/10.3389/fmars.2025.1677103.

Gaston, K.J., O'Neill, M.A., 2004. Automated species identification: why not? Philos. Trans. R. Soc. London [Biol] 359, 655–667. http://dx.doi.org/10.1098/rstb.2003.1442.

Gomes-Pereira, J.N., Auger, V., Beisiegel, K., Benjamin, R., Bergmann, M., Bowden, D., Buhl-Mortensen, P., De Leo, F.C., Dionísio, G., Durden, J.M., Edwards, L., Friedman, A., Greinert, J., Jacobsen-Stout, N., Lerner, S., Leslie, M., Nattkemper, T.W., Sameoto, J.A., Schoening, T., Schouten, R., Seager, J., Singh, H., Soubigou, O., Tojeira, I., van den Beld, I., Dias, F., Tempera, F., Santos, R.S., 2016. Current and future trends in marine image annotation software. Prog. Oceanogr. 149, 106–120. http://dx.doi.org/10.1016/j.pocean.2016.07.005, URL https://www.sciencedirect.com/science/article/pii/S0079661116301240.

Gonzalez-Rivero, M., Kennedy, E., Wyatt, M., Crossman, D., Schaffelke, B., Wachenfeld, D., 2025. ReefCloud: Harnessing AI for integrated coral reef monitoring and agile conservation. In: One Ocean Science Congress 2025. Nice, France, pp. OOS2025–941. http://dx.doi.org/10.5194/oos2025-941.

Günzel, L., 2023. Squidle_ml_algorithm. https://github.com/leoguen/Squidle_ML_Algorithm/. (Accessed 10 Feburary 2026),

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., Lew, M.S., 2016. Deep learning for visual understanding: A review. Neurocomputing 187, 27–48. http://dx.doi.org/10.1016/j.neucom.2015.09.116, Recent Developments on Deep Big Vision, URL https://www.sciencedirect.com/science/article/pii/S0925231215017634.

Hassan, E., Shams, M.Y., Hikal, N.A., Elmougy, S., 2022. The effect of choosing optimizer algorithms to improve computer vision tasks: A comparative study. Multimedia Tools Appl. 82 (11), 16591–16633. http://dx.doi.org/10.1007/s11042-022-13820-0.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep residual learning for image recognition. arXiv:1512.03385, URL https://arxiv.org/abs/1512.03385.

Hedge, P., Souter, D., Jordan, A., Trebilco, R., Ward, T., van Ruth, P., Cowlishaw, M., Lara-Lopez, A., Barrett, N., Thornborough, K., Nichol, S., Przeslawski, R., Parr, A., Kendrick, G., Holmes, T., Pattiaratchi, C., Ferns, L., 2022. Establishing and supporting a national marine baselines and monitoring program: Advice from the marine baselines and monitoring working group. http://dx.doi.org/10.13140/RG.2.2.16944.23043, Prepared for Australia's National Marine Science Committee.

Hernandez, D., Kaplan, J., Henighan, T., McCandlish, S., 2021. Scaling laws for transfer. arXiv:2102.01293, URL https://arxiv.org/abs/2102.01293.

Hinton, G., 2012. Neural networks for machine learning – lecture 6: Overview of mini-batch gradient descent. https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf, Lecture slides, University of Toronto.

Jackett, C., Althaus, F., Maguire, K., Farazi, M., Scoulding, B., Untiedt, C., Ryan, T., Shanks, P., Brodie, P., Williams, A., et al., 2023. A benthic substrate classification method for seabed images using deep learning: Application to management of deep-sea coral reefs. J. Appl. Ecol. http://dx.doi.org/10.1111/1365-2664.14408.

Katija, K., Orenstein, E., Schlining, B., Lundsten, L., Barnard, K., Sainz, G., Boulais, O., Cromwell, M., Butler, E., Woodward, B., et al., 2022. FathomNet: A global image database for enabling artificial intelligence in the ocean. Sci. Rep. 12 (1), http://dx.doi.org/10.1038/s41598-022-19939-2.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Kornblith, S., Shlens, J., Le, Q.V., 2019. Do better ImageNet models transfer better? In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 2656–2666. http://dx.doi.org/10.1109/CVPR.2019.00277.

Krumhansl, K.A., Okamoto, D.K., Rassweiler, A., Novak, M., Bolton, J.J., Cavanaugh, K.C., Connell, S.D., Johnson, C.R., Konar, B., Ling, S.D., Micheli, F., Norderhaug, K.M., Pérez-Matus, A., Sousa-Pinto, I., Reed, D.C., Salomon, A.K., Shears, N.T., Wernberg, T., Anderson, R.J., Barrett, N.S., Buschmann, A.H., Carr, M.H., Caselle, J.E., Derrien-Courtel, S., Edgar, G.J., Edwards, M., Estes, J.A., Goodwin, C., Kenner, M.C., Kushner, D.J., Moy, F.E., Nunn, J., Steneck, R.S., Vásquez, J., Watson, J., Witman, J.D., Byrnes, J.E.K., 2016. Global patterns of kelp forest change over the past half-century. Proc. Natl. Acad. Sci. 113 (48), 13785–13790. http://dx.doi.org/10.1073/pnas.1606102113, https://www.pnas.org/doi/abs/10.1073/pnas.1606102113.

Langenkämper, D., Zurowietz, M., Schoening, T., Nattkemper, T.W., 2017. BIIGLE 2.0 - browsing and annotating large marine image collections. Front. Mar. Sci. 4, http://dx.doi.org/10.3389/fmars.2017.00083, URL https://www.frontiersin.org/articles/10.3389/fmars.2017.00083.

Langlois, L.A., Collier, C.J., McKenzie, L.J., 2023. Subtidal seagrass detector: Development of a deep learning seagrass detection and classification model for seagrass presence and density in diverse habitats from underwater photoquadrats. Front. Mar. Sci. 10, http://dx.doi.org/10.3389/fmars.2023.1197695.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022. http://dx.doi.org/10.1109/ICCV48922.2021.00986.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., Xie, S., 2022. A ConvNet for the 2020s. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11976–11986. http://dx.doi.org/10.48550/arXiv.2201.03545.

Loshchilov, I., Hutter, F., 2019. Decoupled weight decay regularization. arXiv:1711.05101.

MacLeod, N., Benfield, M., Culverhouse, P., 2010. Time to automate identification. Nature 467, 154–155. http://dx.doi.org/10.1038/467154a.

Mahmood, A., Bennamoun, M., An, S., Sohel, F., Boussaid, F., Hovey, R., Kendrick, G., Fisher, R., 2016a. Automatic annotation of coral reefs using deep learning. In: OCEANS 2016 MTS/IEEE Monterey. pp. 1–5. http://dx.doi.org/10.1109/OCEANS.2016.7761105.

Mahmood, A., Bennamoun, M., An, S., Sohel, F., Boussaid, F., Hovey, R., Kendrick, G., Fisher, R.B., 2016b. Coral classification with hybrid feature representations. In: 2016 IEEE International Conference on Image Processing. ICIP, pp. 519–523. http://dx.doi.org/10.1109/ICIP.2016.7532411.

Mahmood, A., Bennamoun, M., An, S., Sohel, F.A., Boussaid, F., Hovey, R., Kendrick, G.A., Fisher, R.B., 2019. Deep image representations for coral image classification. IEEE J. Ocean. Eng. 44 (1), 121–131. http://dx.doi.org/10.1109/joe.2017.2786878.

Mahmood, A., Ospina, A.G., Bennamoun, M., An, S., Sohel, F., Boussaid, F., Hovey, R., Fisher, R.B., Kendrick, G.A., 2020. Automatic hierarchical classification of kelps using deep residual features. Sensors 20 (2), 447. http://dx.doi.org/10.3390/s20020447.

Moltmann, T., Turton, J., Zhang, H.-M., Nolan, G., Gouldman, C., Griesbauer, L., Willis, Z., Piniella, Á.M., Barrell, S., Andersson, E., Gallage, C., Charpentier, E., Belbeoch, M., Poli, P., Rea, A., Burger, E.F., Legler, D.M., Lumpkin, R., Meinig, C., O'Brien, K., Saha, K., Sutton, A., Zhang, D., Zhang, Y., 2019. A global ocean observing system (GOOS), delivered through enhanced Collaboration Across Regions, communities, and new technologies. Front. Mar. Sci. 6, http://dx.doi.org/10.3389/fmars.2019.00291, URL https://www.frontiersin.org/articles/10.3389/fmars.2019.00291.

Moniruzzaman, M., Islam, S.M.S., Bennamoun, M., Lavery, P., 2017. Deep learning on underwater marine object detection: A survey. In: Lecture Notes in Computer Science. pp. 150–160. http://dx.doi.org/10.1007/978-3-319-70353-4_13.

Monk, J., Barrett, N., Bridge, T., Carroll, A., Friedman, A., Ierodiaconou, D., Jordan, A., Kendrick, G., Lucieer, V., 2024. Marine sampling field manual for autonomous underwater vehicles (AUVs). In: Przeslawski, R., Foster, S. (Eds.), Field Manuals for Marine Sampling To Monitor Australian Waters, Version 3. National Environmental Science Program (NESP), URL https://auv-field-manual.github.io/.

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J.V., Lakshminarayanan, B., Snoek, J., 2019. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. 32, http://dx.doi.org/10.48550/arXiv.1906.02530,

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., 2019. PyTorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems 32, http://dx.doi.org/10.48550/arXiv.1912.01703.

Paul, S., Chen, P.-Y., 2022. Vision transformers are robust learners. In: Proceedings of the AAAI conference on Artificial Intelligence. 36, (2), pp. 2071–2081. http://dx.doi.org/10.1609/aaai.v36i2.20103.

Perkins, N.R., Foster, S.D., Hill, N.A., Barrett, N.S., 2016. Image subsampling and point scoring approaches for large-scale marine benthic monitoring programs. Estuar. Coast. Shelf Sci. 176, 36–46. http://dx.doi.org/10.1016/j.ecss.2016.04.005.

Perkins, N., Zhang, Z., Monk, J., Barrett, N., 2022. The annotation approach used for marine imagery impacts the detection of temporal trends in seafloor biota. Ecol. Indic. 140, 109029. http://dx.doi.org/10.1016/j.ecolind.2022.109029, URL https://www.sciencedirect.com/science/article/pii/S1470160X22005003.

Purser, A., Bergmann, M., Lundålv, T., Ontrup, J., Nattkemper, T.W., 2009. Use of machine-learning algorithms for the automated detection of cold-water coral habitats: a pilot study. Mar. Ecol. Prog. Ser. 397, 241–251. http://dx.doi.org/10.3354/meps08154.

Richards, B.L., Beijbom, O., Campbell, M.D., Clarke, M.E., Cutter, G., Dawkins, M., Edington, D., Hart, D.R., Hill, M.C., Hoogs, A., Kriegman, D., Moreland, E.E., Oliver, T.A., Michaels, W.L., Placentino, M., Rollo, A.K., Thompson, C.H., Wallace, F., Williams, I.D., Williams, K., 2019. Automated analysis of underwater imagery: Accomplishments, products, and vision. NOAA Tech. Memo. NMFS-PIFSC-83 http://dx.doi.org/10.25923/0cwf-4714, URL https://repository.library.noaa.gov/view/noaa/20234.

Schneider, N., Piewak, F., Stiller, C., Franke, U., 2017. RegNet: Multimodal sensor registration using deep neural networks. In: 2017 IEEE intelligent vehicles symposium (IV). pp. 1803–1810.

Schoening, T., Durden, J.M., Faber, C., Felden, J., Heger, K., Hoving, H.-J.T., Kiko, R., Köser, K., Krämmer, C., Kwasnitschka, T., et al., 2022. Making marine image data fair. Sci. Data 9 (1), http://dx.doi.org/10.1038/s41597-022-01491-3.

Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. J. Big Data 6 (1), http://dx.doi.org/10.1186/s40537-019-0197-0.

Smith, L.N., 2017. Cyclical learning rates for training neural networks. In: 2017 IEEE Winter Conference on Applications of Computer Vision. WACV, http://dx.doi.org/10.1109/wacv.2017.58.

Stokes, M.D., Deane, G.B., 2009. Automated processing of coral reef benthic images. Limnol. Ocean.: Methods 7 (2), 157–168. http://dx.doi.org/10.4319/lom.2009.7.157.

Sutskever, I., Martens, J., Dahl, G., Hinton, G., 2013. On the importance of initialization and momentum in deep learning. In: International Conference on Machine Learning. pmlr, pp. 1139–1147.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9. http://dx.doi.org/10.1109/CVPR.2015.7298594.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826. http://dx.doi.org/10.1109/CVPR.2016.308.

Tan, M., Langenkämper, D., Nattkemper, T.W., 2022. The impact of data augmentations on deep learning-based marine object classification in benthic image transects. Sensors 22 (14), http://dx.doi.org/10.3390/s22145383, URL https://www.mdpi.com/1424-8220/22/14/5383.

Tan, M., Le, Q.V., 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In: International conference on machine learning. PMLR, pp. 6105–6114.

Taormina, B., Marzloff, M.P., Desroy, N., Caisey, X., Dugornay, O., Metral Thiesse, E., Tancray, A., Carlier, A., 2020. Optimizing image-based protocol to monitor macroepibenthic communities colonizing artificial structures. ICES J. Mar. Sci. 77 (2), 835–845. http://dx.doi.org/10.1093/icesjms/fsz249.

Trotter, C., Griffiths, H.J., Whittle, R.J., 2025. Surveying the deep: A review of computer vision in the benthos. Ecol. Informatics 86, 102989. http://dx.doi.org/10.1016/j.ecoinf.2024.102989, URL https://www.sciencedirect.com/science/article/pii/S1574954124005314.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is all you need. 30, http://dx.doi.org/10.48550/arXiv.1706.03762,

Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K., 2017. Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1492–1500. http://dx.doi.org/10.1109/CVPR.2017.634.