Faster R-CNN Based Deep Learning for Seagrass Detection from Underwater Digital Images

MD Moniruzzaman*, Syed Mohammed Shamsul Islam*†, Paul Lavery* and Mohammed Bennamoun†
*School of Science
Edith Cowan University, Joondalup, WA 6027, Australia
Emails: mmoniruz@our.ecu.edu.au, {syed.islam, p.lavery}@ecu.edu.au
†Department of Computer Science and Software Engineering
University of Western Australia, Crawley, WA 6009, Australia
Email: {syed.islam,mohammed.bennamoun}@uwa.edu.au

Abstract—**Deep learning-based techniques have gained unprecedented success for object detection tasks. The state of the art object detection accuracy and robustness have been achieved by Faster R-CNN framework based algorithms. However, no attempts have been made to detect seagrasses from underwater images mostly due to lack of labelled ground truth dataset, and additional challenges imposed by underwater photographs and low boundary differences among the seagrass and surrounding vegetation. We have created a dataset consisting of 2,699 underwater images of *Halophila ovalis* (one of the common type of seagrasses from Indo-Pacific saltwater environments [1]). We have labelled the seagrass and implemented Faster R-CNN based object detector to detect them from underwater images. We have used Inception V2 network in the Faster R-CNN pipeline and found, this network showed a high mean average precision (mAP) of 0.3464 on laboratory images only, and 0.261 on a test set consists of both field and laboratory images.**

Index Terms—***Halophila ovalis*, Seagrass, Object detection, Faster R-CNN, Inception V2, mAP.**

I. Introduction

Seagrasses are found in intertidal and subtidal marine waters and provide a wide range of important ecological services, including stabilising the seafloor and providing sources of food and habitat for marine invertebrate and vertebrate species [2]. Therefore, monitoring seagrass meadows and their health can help to asses the marine eco-system health [3]–[6].

The first stage of seagrass monitoring is detection and mapping from different image types such as satellite remote sensor based spectral images [7], [8], acoustic images [9], underwater video images [10], spatial [11], [12] and underwater digital images [13], [14]. For close monitoring of seagrass distributions, their health, change of percentage coverage overtimes, underwater digital images are preferred by the marine ecology research community [15].

Machine learning-based approaches are widely used for seagrass classification. Yamamura et al. [16] used otsu classifier, Pizarro et al. [17] used bag of features based maximum likelihood classifier (MLC), Massot-Campos et al. [18] used logistic model tree (LMT), random forest (RF), and multi-layer perceptron (MP), Jalali et al. [14] used scale-invariant feature transform (SIFT), support vector machine (SVM), hierarchical max (HMAX), and colour-quantisation hierarchical max (CQ-HMAX), and Gonzalez Cid et al. [19] & Burguera et al. [20] used SVM classifiers. However, all of these classifiers are shallow machine learning-based classifiers, which are semi-automatic and rely on handcrafted features. Moreover, all the approaches were either coverage estimation [16] or mere classification of seagrass patches from surrounding environment [14], [17]–[19] not detection of specific seagrass species from the underwater image frames.

In recent years, deep learning has created a significant improvement in object detection and classification tasks for hierarchical feature extraction and feature learning [21], [22]. Deep learning has successfully used for classification and object detection of marine habitats (fish [23], [24], planktons [25]–[28], corals [29], [30]) from underwater images. So far deep learning techniques have not been used for seagrass detection or classification from any dataset. In this paper, we describe a method of detecting *Halophila ovalis* (a widely distributed seagrass of ecological significance) from underwater images using Faster R-CNN (faster region-based convolutional neural network) [31]. The main contributions of this paper are:

- Creation of a seagrass (*Halophila ovalis*) underwater image dataset with expert annotation.
- Finding a CNN based Faster R-CNN model for *Halophila ovalis* detection.
- Training the model to create a *Halophila ovalis* detector and evaluating its performance.

The rest of the paper is organised as follows. The background of deep neural networks, especially R-CNN networks, are described in section II. Section III describes the methodology of the research work, section IV illustrates the results and discussions, and finally, section V draws a conclusion and provides future research directions.

II. Background

For the last half-century or more, researchers aimed to allow computers and computer-based systems to model our world well enough to exhibit what is called intelligence [32]. To be intelligent and to act smart enough, computers need to learn from a large amount of data around the world

This research work is supported by Australian government research training program scholarship, and ECU School of Science 2017 Strategic Initiative Fund and Collaborative Research Grant Scheme 2019.

implicitly or explicitly just like the human race learn from surrounding environments. Machine learning, especially deep machine learning architectures have become the matrix for the computer to be intelligent. Deep learning, which is also known as deep structured learning, hierarchical learning or deep machine learning is an artificial neural network (ANN) based system with multiple hidden layers of units between the input and output layers, where this extra layers enable composition of features from lower layers, giving the potential of modelling complex data with fewer units than a similarly performing shallow network [32], [33]. For this reason, deep ANNs have gained unprecedented success in digital image processing [32].

A. *Faster R-CNN Based Object Detection*

The latest addition to the CNN based object detection technique after R-CNN and Fast R-CNN is named Faster R-CNN. Fast R-CNN is faster than R-CNN but was not able to achieve real-time detection for video data using deep neural networks. Both R-CNN and Fast R-CNN are based on the concept of calculating region proposals which act as the tailback. Finding regions of interest in both R-CNN and Fast R-CNN depends on selective search method that uses the greedy algorithm to merge superpixels based on low-level features, which is a slow operation [34]. To overcome this tailback, Ren et al. [35] proposed region proposal networks (RPN) that share CNN layers with the same network for object detection [36]. Overview of object detection with Faster R-CNN has been illustrated in Figure 1.

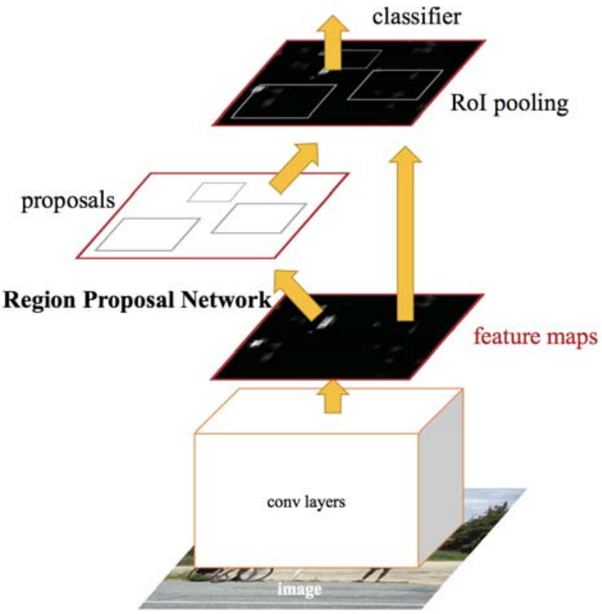


Fig. 1. An overview of object detection with Faster R-CNN. It is a single deep network having two parts: RPN and classifier (taken from [35]).

A Faster R-CNN object detection network consists of two parts: a fully convolutional deep neural network named as RPN whose task is to propose regions of interests and a Fast R-CNN detector [37] to classify the regions. Instead of using the selective search method, an entire image of any size is fed to the RPN. As an output, a set of the proposed region and their objectness score is found [38]. The core idea of Faster R-CNN is to create a shared CNN to avoid the two-stage detection technique. Therefore, the RPN is created in Faster R-CNN by adding some extra layers to the CNN of Fast R-CNN architecture which performs regression simultaneously to produce region proposal and the objectness score. For the generation of region proposals from the convolutional feature map, the RPN uses spatial window sliding technique. The features are then fed through a box-regression layer and box classification layer. For every sliding window location, RPN predicts more than one region proposals. RoI pooling layer of the network afterwards reshape the proposal boxes before classification. Classification layers also predict the bounding box offset values [35].

B. *Faster R-CNN Based on Inception V2*

Though the initial Faster R-CNN was based on VGG16 convolutional neural network, the detection accuracy have been improved due to the extensive experimentation and crafting of the architecture of the CNN's and by increasing the width and height of the network. On the series of improvement for the object detection network, the new network code-named 'inception' has gained the best and highest accuracy in ILSVRC 2014. The name of this network is inspired by the architecture named "network in network (NIN)" designed by Lin et al. [39] and the internet meme "we need to go deeper". However, if the network depth increases, it drastically increases the computational requirements and time. Whenever two networks are chained together due to the increase of their filter, their computation requirements increase quadratic way [40]. This problem can be addressed by introducing sparsity to the network and replacing the fully connected layers with sparse ones [40], [41].

The whole architecture of a Faster R-CNN is built upon the blocks of inception modules. There are two different types of inception modules: the naive version (Figure 2) and the dimension reduction version (Figure 3). While stacking up the naive inception modules on top of each other, the dimension of the networks increases significantly and make the architecture expensive. After adding the pooling layers to the network, the issue becomes even severe. This drawback has been solved by the global use of dimension reduction technique before the convolution operation with larger image patch. Thus the network becomes even more extended and broader without increasing computational complexities. Inception architecture improves the speed three to ten times faster than other non-inception architecture [40].

Inception v2 has been designed to increase the stability of the network towards variations.Table I outlines the layers and architecture of Inception V2. In V2, the conventional inception module has been replaced by a slightly different set of modules

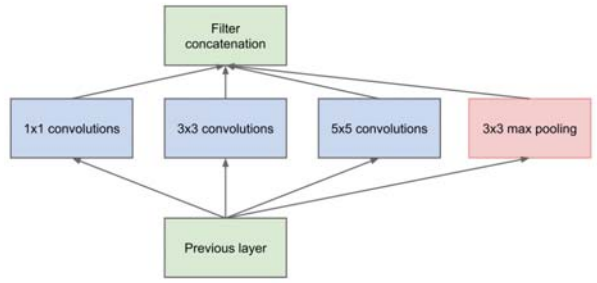


Fig. 2. The naive version of the Inception module (taken from [40]).

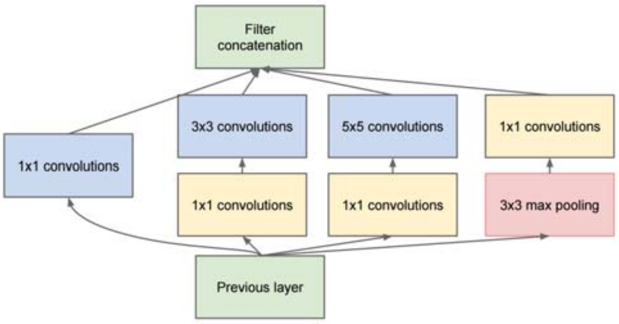


Fig. 3. Inception module with dimension reduction (taken from [40]).

described in [42]. This upgraded inception architecture has performed well in coco dataset with an mAP of 0.28.

TABLE I
INCEPTION V2 ARCHITECTURE DESCRIBED IN [42]

Type	Patch Size	Input size
conv	3x3/2	299x299x3
conv	3x3/1	149x149x32
conv padded	3x3/1	147x147x32
pool	3x3/2	147x147x64
conv	3x3/1	73x73x64
conv	3x3/2	71x71x80
conv	3x3/1	35x35x192
3xInception	N/A	35x35x288
5xInception	N/A	17x17x768
2xInception	N/A	8x8x1280
pool	8x8	8x8x2048
linear	logits	1x1x2048
softmax	classifier	1x1x1000

III. METHODOLOGY

The proposed approach for *Halophila ovalis* detection is performed in three steps: data collection and pre-processing network selection and training, and finally, evaluation of the model. All these steps illustrated in Figure 4 are discussed as follows.

A. Data collection

We have collected underwater images of *Halophila ovalis* growing in shallow coastal waters of Western Australia and an experimental facility at Edith Cowan University. Photographs were taken using a Fujifilm X-T30 mirror-less digital camera with XF 18-55mm F 2.8-4 R LM OIS lens. The camera with the lens was encapsulated inside a compatible underwater housing. Most of the underwater images from the natural shorelines were collected from Dampier reserve, Mandurah, Fremantle and Rottness island. Some images were collected from Coral Bay and Exmouth areas. A small number of images were collected from Lucky Bay, Milyu, Pelican Point and Rocky Bay of Swan Canning area of Western Australia. All of those data collection tasks were organised and performed with the help of the Centre for Marine Ecosystems Research, Edith Cowan University. A total of 499 images were selected from all the field surveys to prepare the training and testing datasets. The remainder of the images were discarded due to the absence of *Halophila ovalis*, extreme blurriness, or low to zero visibility.

A total of 2,200 images were collected from the aquaria at the greenhouse facility. As the water inside the laboratory is free from natural sedimentation, sun glint and microparticles, these images are brighter than the natural images. However, as the depth of the water was around six inches only, this close distance of the plants from the camera lens created a challenge of blurriness. These lab images are helpful to train the network to recognise and detect *Halophila ovalis* in optimal condition and also helpful to retrain on natural and real-life images. Our final dataset contains a total of 2,699 images, which is a combination of underwater images both from under laboratory condition and real-life situation. We named this data set ECU *Halophila ovalis*-1 (ECUHO-1). Some of the examples of the datasets are shown in Figure 5.

B. Data labelling

All the images from both real and laboratory environments are then labelled using 'LabelImg' software which is a python based graphical image annotation tool. LabelImg uses Qt (an open-source widget toolkit) to create a graphical user interface. All the images are saved as 'XML' files with PASCAL VOC format. The labelling interface is demonstrated at Figure 6. This dataset can be treated as ground truth for underwater *Halophila ovalis* images and can be used for future research purposes.

C. Training Faster R-CNN Inception V2 Detector

For training the detector, we have divided the whole ECUHO-1 dataset into the training set and testing set. ECUHO-1 training set consists of 2,160 images, and the testing set consists of 539 images (which is approximately 20% of the whole dataset). We built a separate dataset of a total of 369 images containing 209 real (training set) and 160 laboratory images (testing set). We named this second dataset as ECUHO-2. Inception V2 based Faster R-CNN detector was trained in two separate training sessions using the training set from ECUHO-1 and ECUHO-2. During the training session with ECUHO-2, 2986 regions of interest (labelled bounding box containing *Halophila ovalis*) were fed to the network.

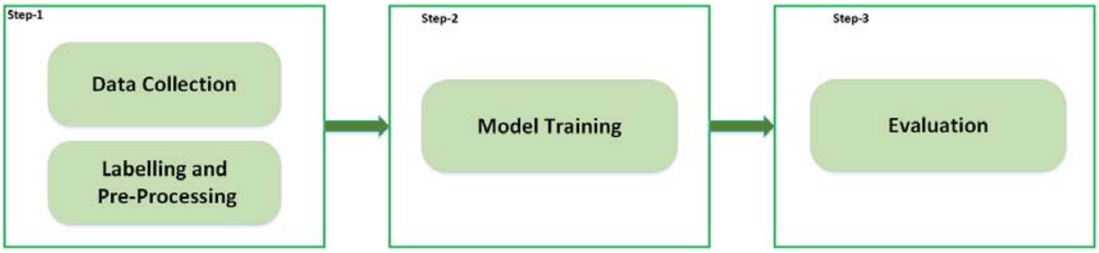


Fig. 4. Block diagram of the proposed methodology

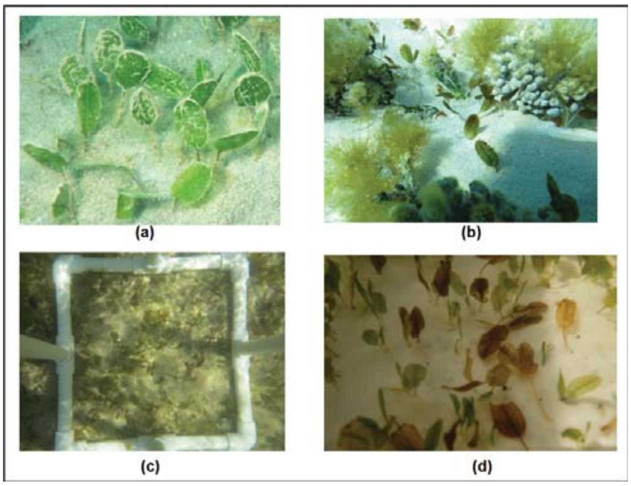


Fig. 5. Sample images from ECUHO-1: (a), (b), (c) are from the images from natural environment, (d) is taken from laboratory environment

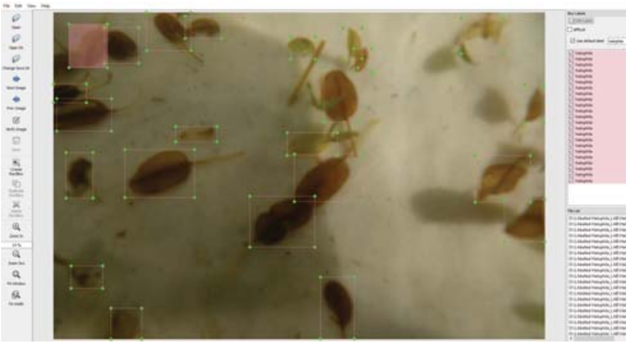


Fig. 6. *Halophila ovalis* labeling using labelImg tool

While training with the training set of ECUHO-1, 95,959 regions of interest were used. While feeding the images to the network, all the images were re-sized to 1024x600 dimension.

For both the training sessions, at the initial input layer, the height and width stride was kept 16. At maxpooling layer, the kernel size was kept 2 with stride 2. For training, the batch size was set to only one as the dataset contains images mostly with high dimensions (6000x4000). Learning rate for the training was set to 0.0002 with a total number of 200000 steps to finish the training. The random horizontal flip operation was performed to augment the image dataset, which helped to increase the training and test accuracy and precision.

The training losses are listed for both the training sessions of the detector in Table II. We can see the training loss for the training session with the training set of ECUHO-1 is higher than the training set of ECUHO-2 as the containing images have significant differences in terms of clarity, size, shape and colour of objects in the images. Moreover, the laboratory images are affected by blurriness due to the closeness of *Halophila ovalis* plants from the camera. So, while adjusting and sharing the weights, the training loss grew higher.

Figure 7 visualises all the training loss graphs for the training session with ECUHO-2 training set. Throughout the whole training period, the classification loss, classifier localisation loss and RPN objectness loss dropped consistently. At the end of the training, after 200k steps, the training classification loss was 0.02, classifier localisation loss was below 0.02, and RPN objectness loss was 0.005. The RPN localisation loss fluctuated, but at the end of the training, it dropped to 0.04. Also, the total training loss was less than 0.1.

Figure 8 shows all the training loss graphs for field images (ECUHO-1 dataset). The Final training loss, box classifier loss, classifier localisation loss, RPN localisation loss, and objectness loss are 0.91, 0.27, 0.34, 0.24, and 0.047 respectively.

Training time required for each step varied from 3.281 seconds to 3.176 seconds using single GPU of an 8 GPU deep learning server called 'Lambda Blade' containing 8 NVIDIA GTX 1080 Ti graphics processing units.

TABLE II
LOSSES FOR TRAINING ON ECUHO-1 & 2 DATASETS

Loss Type	**Loss Value**	
-	ECUHO-1	ECUHO-2
Final total loss	0.9153	0.10
Final classifier classification loss	0.2761	0.02
Final classifier localization loss	0.3446	0.0201
Final RPN localization loss	0.24	0.04
Final RPN objectness loss	0.0473	0.0052

D. Performance Evaluation

Once the training was completed, the inception V2 based Faster R-CNN object detector was tested and evaluated using COCO detection metrics [43]. COCO detection metrics have been developed to score the models participating in COCO competition. Despite having several similarities with the Pascal

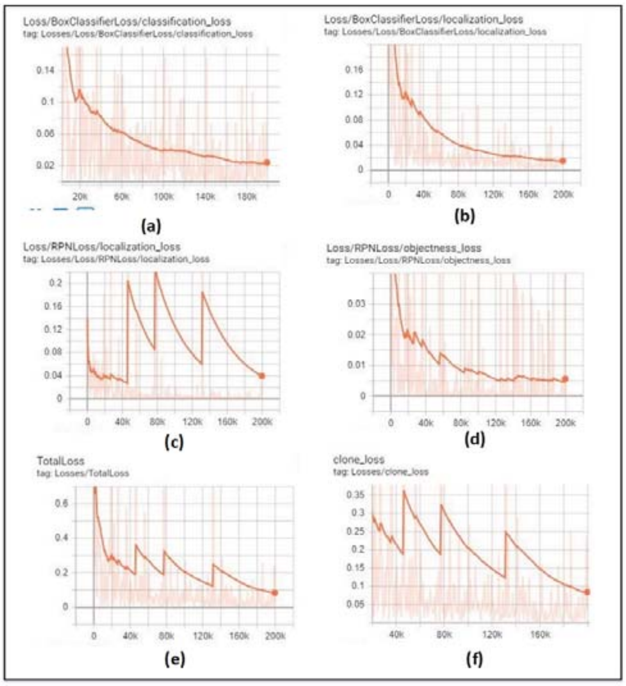


Fig. 7. Loss curves of training Inception V2 based Faster R-CNN detector with training set of ECUHO-2: (a) Classification loss, (b) Classifier localisation loss, (c) RPN localization loss, (d) RPN objectness loss, (e) Total loss and (f) clone loss

VOC [44], COCO metrics provides with additional pieces of information such as mAP values at IoU (intersection over union) ranging from 0.50 to 0.95, recalls and precisions for large, medium, and smaller values and so on. For the evaluation purpose, we used two separate testing sets from ECUHO-1 and ECUHO-2. ECUHO-1 testing set contains 539 images from both field and laboratory collected images, and ECUHO-2 testing set contains 160 images from the laboratory facility. Both the testing sets were labelled with expert advice and worked as ground truth for this evaluation task. The total number of regions of interest in the ECUHO-1 testing set is 20,215, which was used to calculate precision and recall.

The ability of an object detector to identify only the target object is called its precision. It is the percentage of the correct positive prediction over the total detection. On the other hand, recall is the percentage of the detected true positives with respect to all the ground truths. It is the detectors ability to find all the target objects. The mathematical expression of precision and recall are (as described in [45]):

$$Precision = \frac{TruePositive(TP)}{TruePositive(TP) + FalsePositive(FP)} \tag{1}$$

$$Recall = \frac{TruePositive(TP)}{TruePositive(TP) + FalseNegative(FN)} \tag{2}$$

Here, true positive (TP) is the value of a correct detection and it is denoted by $IoU \geq threshholdvalue$. False positive (FP) is a incorrect detection where $IoU \leq threshholdvalue$. The third important parameter is a false negative, which is a ground truth bounding box that has not been detected.

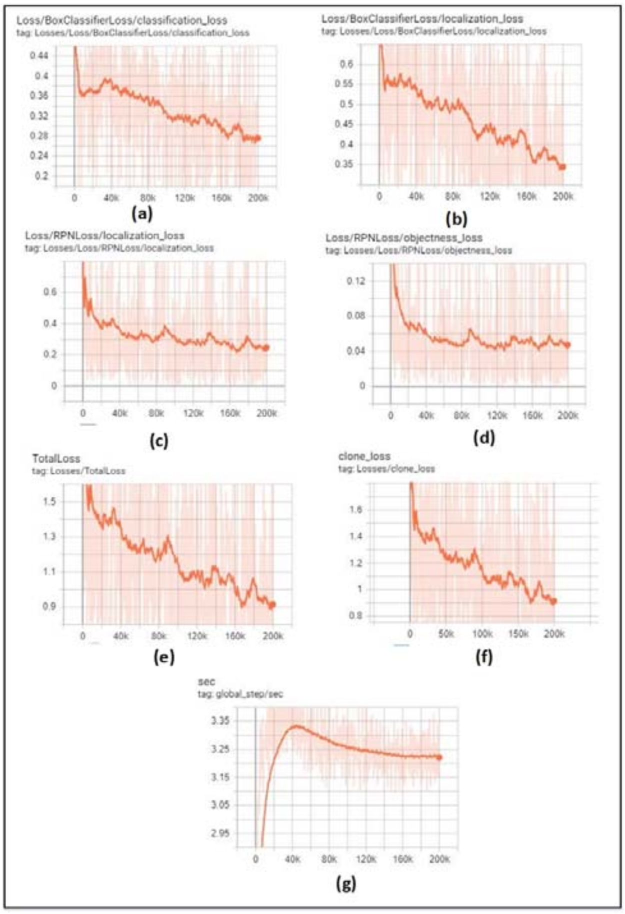


Fig. 8. Loss curves of training the Inception V2 based faster R-CNN model with ECUHO-1: (a) Classification loss, (b) Classifier localisation loss, (c) RPN localization loss, (d) RPN objectness loss, (e) Total loss, (f) clone loss, and (g) time required per step

IV. Result and Discussion

Table III shows the precision and recall values of Inception V2 based Faster R-CNN object detection model. From the table, we can see that while testing the model trained with ECUHO-1 with the corresponding testing set, the mean average precision is 0.26 (which is close to 0.28 that was achieved by the Faster R-CNN Inception V2 object detector over COCO dataset). For IoU threshold of 0.50, the precision increased to 0.668. Average recall over 100 objects in an image frame is 0.357, for ten objects, it is 0.129.

For the testing set of ECUHO-2, the average mean precision of the detector is 0.3464 (which is even more than the precision achieved by the same network on coco dataset). If the IoU [46] threshold is dropped to 0.50, the mean average precision increases to 0.76. The recall value for over 100 objects in an image frame is 0.47.

For the ECUHO-1 data set, the higher number of training and testing images are collected from laboratory tanks that suffer from low water depths. So the images are affected by blurriness. This blurriness caused unexpected noise to the dataset and resulted in higher training loss and lower mAP for ECUHO-1. Moreover, the number of total images are inadequate to train a deep network like Inception V2 and provides an optimal detection accuracy.

TABLE III
EVALUATING INCEPTION V2 WITH TESTING SET OF ECUHO-1 AND ECUHO-2

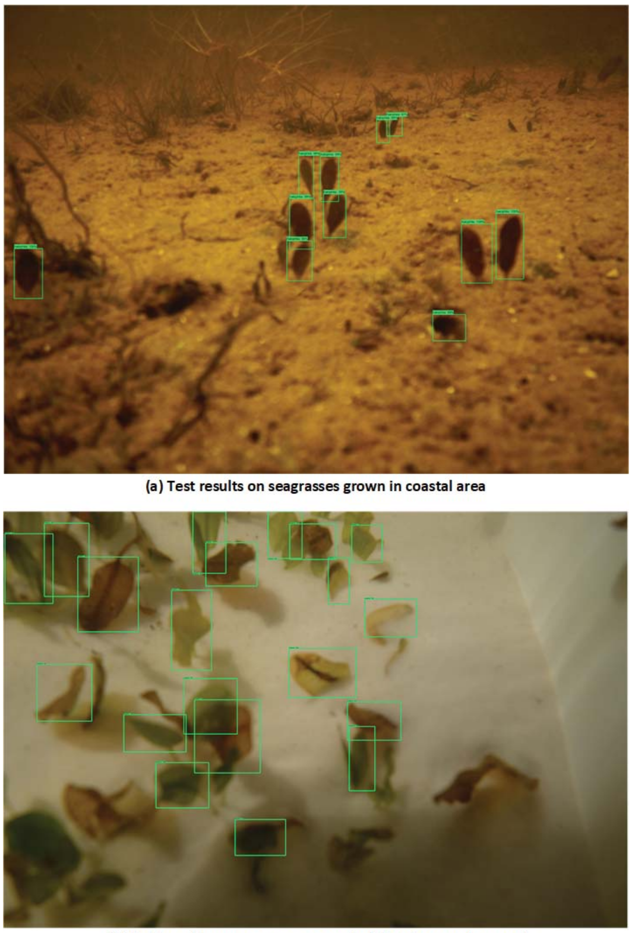
Evaluating Parameters	Value	
-	ECUHO-1	ECUHO-2
Detection Boxes Precision (mAP)	0.261	0.3464
Detection Boxes Precision (mAP)@.50IOU	0.668	0.7663
Detection Boxes Precision (mAP)@.75IOU	0.142	0.2378
Detection Boxes Recall (AR)@1	0.015	0.3004
Detection Boxes Recall (AR)@10	0.129	0.2465
Detection Boxes Recall (AR)@100	0.357	0.4699

Figure 9 presents the visual representation of the detection of *Halophila ovalis*. Figure 9 (a) is a test image collected from the real-life environment and (b) is a laboratory image. Both the images show considerably good detection accuracy and precision although a few of the *Halophila* were not detected in (b) and the detector missed a couple of leaves at the far back challenged by the water turbidity.

Total time required for evaluating 539 images was only 55.85 seconds, which is very fast and indicates that the detector can be used for video data in real-time using a similar system used for training.

V. Conclusion and Future Work

In this paper, we have proposed Inception V2 based Faster R-CNN network to detect *Halophila ovalis* from underwater image dataset. Results show that the proposed model achieve high precision (mAP $0.26 - 0.3464$) and can be used for detection and automatic annotation of *Halophila ovalis*. This model can also be adopted for other seagrasses and leaf-based plant detection and annotation tasks. A complete training set of 2,160 images is a relatively small dataset for a deep network like Inception V2. Moreover, the images from the laboratory suffered from blurriness. So the accuracy can be increased with a larger and clearer image dataset. In future, automatic network architecture search network (NASNet) may be applied to find a suitable architecture for the optimum *Halophila ovalis* and other seagrass detection task. Generative adversarial network (GAN) can be used to generate synthetic images which will increase the size of the dataset without collecting more images.



(a) Test results on seagrasses grown in coastal area

(b) Test results on seagrasses grown in laboratory environment

Fig. 9. *Halophila ovalis* detection using Inception V2 based Faster R-CNN detector

Acknowledgment

The authors acknowledge the contributions of Nicole Said, Chanelle Webster, and Caitlyn O'Dea for the *Halophila ovalis* image collection and expert advice for labelling tasks and Giang Truong for helping with programming issues. This research work has been performed with the help of the Australian government research training program (RTP) scholarship, and ECU School of Science 2017 Strategic Initiative Fund and Collaborative Research Grant Scheme 2019.

References

[1] P. Young and H. Kirkman, "The seagrass communities of moreton bay, queensland," *Aquatic Botany*, vol. 1, pp. 191–202, 1975.

[2] M. A. Hemminga and C. M. Duarte, *Seagrass ecology*. Cambridge University Press, 2000.

[3] C. Lafabrie, C. Pergent-Martini, and G. Pergent, "Metal contamination of posidonia oceanica meadows along the corsican coastline (mediterranean)," *Environmental Pollution*, vol. 151, no. 1, pp. 262–268, 2008.

[4] P. S. Lavery, K. McMahon, J. Weyers, M. C. Boyce, and C. E. Oldham, "Release of dissolved organic carbon from seagrass wrack and its implications for trophic connectivity," *Marine Ecology Progress Series*, vol. 494, pp. 121–133, 2013.

[5] M. A. Lewis, D. D. Dantin, C. A. Chancy, K. C. Abel, and C. G. Lewis, "Florida seagrass habitat evaluation: a comparative survey for chemical quality," *Environmental Pollution*, vol. 146, no. 1, pp. 206–218, 2007.

[6] R. Purvaja, R. Robin, D. Ganguly, G. Hariharan, G. Singh, R. Raghuraman, and R. Ramesh, "Seagrass meadows as proxy for assessment of ecosystem health," *Ocean & coastal management*, vol. 159, pp. 34–45, 2018.

[7] C. Roelfsema, S. Phinn, N. Udy, and P. Maxwell, "An integrated field and remote sensing approach for mapping seagrass cover, moreton bay, australia," *Journal of Spatial Science*, vol. 54, no. 1, pp. 45–62, 2009.

[8] P. J. Mumby and A. J. Edwards, “Mapping marine environments with ikonos imagery: enhanced spatial resolution can deliver greater thematic accuracy,” *Remote sensing of environment*, vol. 82, no. 2-3, pp. 248–257, 2002.

[9] A. Vasilijevic, N. Miskovic, Z. Vukic, and F. Mandic, “Monitoring of seagrass by lightweight auv: A posidonia oceanica case study surrounding murter island of croatia,” in *22nd Mediterranean Conference on Control and Automation*. IEEE, 2014, pp. 758–763.

[10] J. G. Norris, S. Wyllie-Echeverria, T. Mumford, A. Bailey, and T. Turner, “Estimating basal area coverage of subtidal seagrass beds using underwater videography,” *Aquatic Botany*, vol. 58, no. 3-4, pp. 269–287, 1997.

[11] A. V. Uhrin and P. A. Townsend, “Improved seagrass mapping using linear spectral unmixing of aerial photographs,” *Estuarine, Coastal and Shelf Science*, vol. 171, pp. 11–22, 2016.

[12] J. P. Duffy, L. Pratt, K. Anderson, P. E. Land, and J. D. Shutler, “Spatial assessment of intertidal seagrass meadows using optical imaging systems and a lightweight drone,” *Estuarine, Coastal and Shelf Science*, vol. 200, pp. 169–180, 2018.

[13] Y. Gonzalez-Cid, A. Burguera, F. Bonin-Font, and A. Matamoros, “Machine learning and deep learning strategies to identify posidonia meadows in underwater images,” in *OCEANS 2017-Aberdeen*. IEEE, 2017, pp. 1–5.

[14] S. Jalali, P. J. Seekings, C. Tan, H. Z. Tan, J.-H. Lim, and E. A. Taylor, “Classification of marine organisms in underwater images using cq-hmax biologically inspired color approach,” in *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2013, pp. 1–8.

[15] M. Moniruzzaman, S. Islam, P. Lavery, M. Bennamoun, and C. P. Lam, “Imaging and classification techniques for seagrass mapping and monitoring: A comprehensive survey,” *arXiv preprint arXiv:1902.11114*, 2019.

[16] M. Yamamuro, K. Nishimura, K. Kishimoto, K. Nozaki, K. Kato, A. Negishi, K. Otani, H. Shimizu, T. Hayashibara, M. Sano *et al.*, “Mapping tropical seagrass beds with an underwater remotely operated vehicle (rov),” *Recent advances in marine science and technology*, pp. 177–181, 2002.

[17] O. Pizarro, P. Rigby, M. Johnson-Roberson, S. B. Williams, and J. Colquhoun, “Towards image-based marine habitat classification,” in *OCEANS 2008*. IEEE, 2008, pp. 1–7.

[18] M. Massot-Campos, G. Oliver-Codina, L. Ruano-Amengual, and M. Miró-Juliá, “Texture analysis of seabed images: Quantifying the presence of posidonia oceanica at palma bay,” in *2013 MTS/IEEE OCEANS-Bergen*. IEEE, 2013, pp. 1–6.

[19] Y. Gonzalez-Cid, A. Burguera, F. Bonin-Font, and A. Matamoros, “Machine learning and deep learning strategies to identify posidonia meadows in underwater images,” in *OCEANS 2017-Aberdeen*. IEEE, 2017, pp. 1–5.

[20] A. Burguera, F. Bonin-Font, J. L. Lisani, A. B. Petro, and G. Oliver, “Towards automatic visual sea grass detection in underwater areas of ecological interest,” in *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2016, pp. 1–4.

[21] H. A. Song and S.-Y. Lee, “Hierarchical representation using nmf,” in *International conference on neural information processing*. Springer, 2013, pp. 466–473.

[22] M. Moniruzzaman, S. M. S. Islam, M. Bennamoun, and P. Lavery, “Deep learning on underwater marine object detection: a survey,” in *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, 2017, pp. 150–160.

[23] X. Li, M. Shang, H. Qin, and L. Chen, “Fast accurate fish detection and recognition of underwater images with fast r-cnn,” in *OCEANS 2015-MTS/IEEE Washington*. IEEE, 2015, pp. 1–5.

[24] S. Villon, M. Chaumont, G. Subsol, S. Villéger, T. Claverie, and D. Mouillot, “Coral reef fish detection and recognition in underwater videos by supervised machine learning: Comparison between deep learning and hog+ svm methods,” in *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, 2016, pp. 160–171.

[25] O. Py, H. Hong, and S. Zhongzhi, “Plankton classification with deep convolutional neural networks,” in *2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*. IEEE, 2016, pp. 132–136.

[26] H. Lee, M. Park, and J. Kim, “Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning,” in *2016 IEEE international conference on image processing (ICIP)*. IEEE, 2016, pp. 3713–3717.

[27] J. Dai, R. Wang, H. Zheng, G. Ji, and X. Qiao, “Zooplanktonet: Deep convolutional network for zooplankton classification,” in *OCEANS 2016-Shanghai*. IEEE, 2016, pp. 1–6.

[28] S. Dieleman, A. Van den Oord, I. Korshunova, J. Burms, J. Degrave, L. Pigou, and P. Buteneers, “Classifying plankton with deep neural networks,” *Blog entry*, vol. 3, p. 4, 2015.

[29] A. Mahmood, M. Bennamoun, S. An, F. Sohel, F. Boussaid, R. Hovey, G. Kendrick, and R. Fisher, “Coral classification with hybrid feature representations,” in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 519–523.

[30] M. Elawady, “Sparse coral classification using deep convolutional neural networks,” *arXiv preprint arXiv:1511.09067*, 2015.

[31] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.

[32] Y. Bengio *et al.*, “Learning deep architectures for ai,” *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[33] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural networks*, vol. 61, pp. 85–117, 2015.

[34] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, “Selective search for object recognition,” *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[35] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.

[36] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[37] R. Girshick, “Fast r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[38] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[39] M. Lin, Q. Chen, and S. Yan, “Network in network,” *arXiv preprint arXiv:1312.4400*, 2013.

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[41] S. Arora, A. Bhaskara, R. Ge, and T. Ma, “Provable bounds for learning some deep representations,” in *International Conference on Machine Learning*, 2014, pp. 584–592.

[42] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[43] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollár, and C. L. Zitnick, “Microsoft coco captions: Data collection and evaluation server,” *arXiv preprint arXiv:1504.00325*, 2015.

[44] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes challenge: A retrospective,” *International journal of computer vision*, vol. 111, no. 1, pp. 98–136, 2015.

[45] M. C. McJunkin, “Precision and recall in title keyword searches,” *Information Technology and Libraries*, vol. 14, no. 3, p. 161, 1995.

[46] M. A. Rahman and Y. Wang, “Optimizing intersection-over-union in deep neural networks for image segmentation,” in *International symposium on visual computing*. Springer, 2016, pp. 234–244.