Detection and Classification of Corals Fromunder-Sea Images Using Deep Learning

Bhagyashree Patra[1], Dr. Tapaswini Nayak[2]

[1,2] Dept. of Computer Science, MITS School of Biotechnology

Abstract
Coral reefs are considered, an essential part of the marine ecosystem, as they support a large part of marine life. Deep sea exploration and imaging have provided a great opportunity to look into the vast and complex marine ecosystem. Manual annotation is a tedious and time- consuming task for marine experts and takes a lot of time for them to manually annotate a single image. Therefore, automatic detection of corals is important as it helps the experts to keep track of vulnerable coral species in conjunction with other marine life, to detect and annotate them.
However, this is a challenging task, as most of the common coral species look familiar to each other even if they belong to a different category. Additionally, in a single colony, varieties of coral co-exhibit and coral reefs serve as the home of a variety of aquatic species. Since, deep learning procedures of detection and classification are based on photographic images with features like image quality, distance from the image, angle of the object and underwater lighting conditions, this task is challenging.
A large dataset of 2500 images has been taken and is classified into 10 different coral species in the present automated deep-learning coral detection scheme, for detection and classification accuracy. In the current work, the fundamental analysisof these images is done, using techniques of Deep learning (DL) and Convolutional Neural Networks (CNN). To enhance the easiness, efficiency and speed, a known database of corals called structure RSMAS (consisting only 600 images) is used to develop/design a prototype. To enhance the capability of our system, public image data available online at inaturalist.org, with 1000 images has been used, giving an accuracy of 94%. Currently, an existing DL model called as MobileNet (for classification) with an SSD detection head (for detection), is being used. Further, detection and classification could be increased by data cleaning and using high- resolution images and design a customized DL model Architecture.

Keywords: Coral Detection, Structural Coral images, Deep Learning, Transfer Learning, Convolutional Neural Network, Structured RSMAS, MobileNet, SSD.

Abbreviations

RSMAS - Rosenstiel School of Marine and Atmospheric Sciences

KTH-TIPS – KTH-Textures under varying Illumination, Pose, and Scale CURET - Columbia-Utrecht Reflectance and Texture

UIUCtex - University of Illinois at Urbana-Champaign texture 155 datasets MLC - Moorea Labelled Corals

CNN – Convolutional Neural Network DL – Deep Learning

ILSVRC - ImageNet Large Scale Visual Recognition Challenge SSD - Single Shot Multibox Detector

IoU – Intersection over Union NMS – Non-maximum Suppressions ML - Machine Learning

INTRODUCTION

Coral reefs play an important role in the marine ecosystem as well as in human life. These are living organisms consisting of small animals called “polyps”, and exist in more than 200 countries. Coral reefs are found where the sea is shallow (less than 100m); where the sea is warm (usually between 25° and 29°C); and therefore, are located within the latitude of 30°N to 30°S i.e., only in tropical seas. According to Scientists’ predictions, all corals will be threatened by 2050, and 75% will face high to critical threat levels. According to David Obura, chair of the Coral Specialist Group in the International Union for the Conservation of Nature. By 2070, coral reefs could be gone altogether. He made a quote that “Children born today may be the last generation to see coral reefs unless we do everything to limit warming to 1.5C, we will lose 99% of the world’s coral reefs”.

Physical damage like dredging, quarrying, destructive fishing practices and gear, boat anchors and groundings, and recreational misuse affect corals. Pollution that originates on lands like sedimentation, Nutrients (nitrogen and phosphorous) from agricultural and residential fertilizer use, Pathogens from inadequately treated sewage, Toxic substances, including metals, organic chemicals, and pesticides found in industrial and Trash and micro-plastics from improper disposal affects coral reproduction, growth, and other physiological processes. Over-exploitation (Over-fishing), for food, for the aquarium trade, for the trinket trade, for medicinal purposes, Destructive fishing practices, Coral mining (Overexploitation/ Habitat Destruction), Sediment, nutrient and chemical pollution, Marine based pollution, Irresponsible tourism, Global warming and climate change Effects of sea level rise, Effects of more dissolved carbon dioxide, Coral harvesting for the aquarium trade and jewelry cause physical damage to corals as well. Ocean acidification which uptakes CO_2 in oceans is the greatest global threat to coral reef ecosystems leading to coral bleaching and death of corals.

But deep-sea exploration and imaging have provided a great opportunity to look into the vast and complex marine ecosystem. Thousands of images of coral reefs are captured regularly by Autonomous Underwater Vehicle (AUV). Manual annotation is a tedious and time- consuming task for marine experts and takes a lot of time for them to manually annotate a single image, therefore automatic detection of corals is important as it helps the expert to keep track of vulnerable coral species as well as to detect and annotate them. Having a rigorous accurate automatic coral detector can help in analyzing the large amount of data, thereby progressing in the understanding of coral reefs.

Object detection is a basic task in computer vision. In object detection, our main focus is to classify the object as well as to localize it in images or real-time videos. Many object detection methods have been proposed in the past few decades. Early methods mainly rely on handcrafted features and shallow machine learning algorithms (Beijbom et al. 2012; M. Stokes 2009; Pizarro et al. 2008; Shihavuddin et al. 2013) which are easy to overfit. As a result, the performance is not satisfactory. In recent years, deep artificial neural networks have emerged and become the dominant methods in object detection problems, particularl in Computer Vision (Al-Mualla, Basaeed, and Bhaskar 2016; Hinton, Krizhevsky, Sutskever, and 2017; G. Liu et al. 2017). ILSVRC is a clear example of this one (Russakovsky et al. 2014), who’s top ten models were competing in CNNs since 2012. Similarly, there are more common CNN models available such as LeNet(Lecun et al. 1998), VGGNet(Simonyan and Zisserman 2014), InceptionNet(Szegedy et al. 2016), ResNet(He et al. 2016), DenseNet(Huang et al. 2017), and many more which are used before by so many researchers for classifying coral texture images or structure images or both. These models are capable of extracting simple and complex features as the network goes deeper also they are capable of adding a connection to features that skip in other layers. Furthermore, they are able to overcome the limitation of large datasets by increasing the size of the training set artificially by the method of data augmentation or by starting the training using the weights from the pre-trained network on another dataset called transfer learning.

In this work, our main objective is to detect the corals from undersea images and annotate them by the help of a bounding box from photos as well as from real-time videos. However, this is a challenging task, as most of the common coral species looks familiar to each other even if they belong to a different category. Additionally, in a single colony, varieties of coral co-exhibit and coral reefs serve as the home of a variety of aquatic species. Since, this proposed deep learning procedure are based on photographic images, the task is difficult due to image quality, distance from image, angle of the object and underwater lighting conditions. Here we propose the CNN architecture model named as MobileNet for classifying the corals by analyzing the features extracted from the input image, MobileNet uses depth-wise separable convolution method for better performance of the model and then for detection purposes, to generate a bounding box we used a single stage detector model called SSD (W. Liu et al. 2015). In order to achieve this, we have used a structured coral dataset called Structure RSMAS available in this given link: https://sci2s.ugr.es/CNN-coral- image-classification. But the deep-learning method for better detection accuracy required more amount of data, and as Structured RSMAS contains only limited images for supplementary data samples we collect the images from the following link: https://www.inaturalist.org/, to enrich coral morphological information.

Literature review

Previously there is much research performed on coral classification and annotation. And currently, nine open data sources are available for coral research such as KTH-TIPS, CURET, UIUCtex, MLC, EILAT, EILAT2, RSMAS, Structured RSMAS and the Red Sea Mosaic. Only MLC, EILAT, EILAT2, RSMAS, and the Red Sea Mosaic contain the RGB format images. From these, RSMAS and EILAT are the ones with a greater number of classes, 14 and 8, respectively, and the most recent. EILAT2 is a subset of EILAT, MLC 160 only contains five coral classes and the Red Sea Mosaic is a large image containing different coral species. The images in all these datasets share some characteristics inherent to underwater images: the water movement causes lightning variations between images taken at the same time; the water causes the images to be blurry. In our work, we have taken the Structure RSMAS dataset, which contains the structure images of corals which is suitable for our study.

Before recent advancements in deep learning, most previous approaches to automatic coral
reef classification with underwater images combined classical machine learning models with feature extraction algorithms (Beijbom et al. 2012; Marcos, Soriano, and Saloma 2005; M. Stokes 2009; Shihavuddin et al. 2013). They used, an general algorithm to extract color features and texture features. Then, they used them to train a classical machine learning model, like Support Vector Machines in (Beijbom et al. 2012) or a three-layer neural network in (Marcos, Soriano, and Saloma 2005).Most of them used a single dataset, usually small and containing several non-coral classes, except for (Shihavuddin et al. 2013)where the authors designed an algorithm that could be used with different datasets. They divided their algorithm into steps and at each step, several sub-algorithms could be chosen in accordance with the dataset that was going to be classified. The authors in (Ani Brown Mary and Dejey 2018; Dharma and Mary 2017; Russakovsky et al. 2014; Shihavuddin et al. 2013) proposed three different modifications of the local binary pattern feature descriptor, and they used the extracted

features to train machine learning models in order to classify several datasets. In 2015, the author in (Elawady 2015) used CNNs for the first time to resolve this task. He used a LeNet-5 model, but he still combined it with feature extraction algorithms. Since then, most works classifying coral images have used CNNs, as they provide better results and do not need a previous step of feature extraction. The authors in (Dharma 2018) and (Ammar Mahmood et al. 2017) used a VGGNet model and in (Ammar Mahmood et al. 2017) the authors also combined the features extracted by the CNN model with hand-crafted features, although the improvement from such custom features was small. In(Dharma 2018) they 200 used the Benthoz15 dataset and in (Ammar Mahmood et al. 2017) they used the MLC dataset. In (Gómez-Ríos et al. 2019), the authors proposed the use of more powerful CNNs to classify EILAT and RSMAS (A Mahmood et al. 2016), and they found that ResNet was the best CNN architecture in both datasets, improving the state-of-the-art results obtained by (Mary and Dharma 2017; Shihavuddin et al. 2013),(Ani Brown Mary and Dejey 2018), which were all the works that used EILAT and RSMAS. The works that used RSMAS and result is a recall of 99.34%, obtained by (Shihavuddin et al. 2013), but using a held-out test of 10%. The authors in (Gómez-Ríos et al. 2019) obtained an accuracy of 98.63% using a five-fold cross-validation technique, which means that they were using test sets of 20% of the dataset in each partition. Then, they took the mean of 210 the test accuracy on the five partitions. Consequently, this result is more stable. In (Song et al. 2021)the author developed a coral investigation system on RGB and spectral images where they applied CNN on single channel image, using ResNet as a base Network and named it as DeeperLabC and got the F1-Score value as 97.10% but this is a binary classification which classifies the corals and non-corals. In (Raphael et al. 2020) the author used VGG-16 for doing the classification of corals and got an accuracy of 90%, they performed the research on the dataset collected from Gulf of Eilat ($29^0 30'$N, $34^0 55'$ E) which contains only four different kinds of corals. Again, they tested the ResNet-50 CNN model on a dataset containing 11 different kind of corals and got accuracy of 80.13%.

The present work is somewhat different from all the previously cited works in this we develop an automatic model capable of identifying and detecting the species of a coral based on an input image. The current system is able to classify multiple classes of corals as well as localize those corals provided as input images to solve issues like the tedious repetitive task of manual annotation. It also enables rapid and accurate processing of massive data.

Materials And Methods

This prospective comparative study was carried out on different types of corals; we mainly focused on 10 different types of corals. A total of 2500 corals images have been taken for this study. The sample size actually obtained for this study was 250 corals for each group of species and images are in RGB format. The input images are collected from different data sources available online like some of them are collected from https://www.inaturalist.org/, https://universe.roboflow.com/giang-ha/structure-rsmas, and other public resources. 10 corals species have been taken for carrying out this work such as Acropora Palmata, Acropora, Aplysina Fistularis (Sea-sponge), Colpophyllia Natans (Brain-Corals), Diadema Savignyi, Meandrina Meandrites, Melithaea Rubra (precious-Corals), Montastraea cavernosa, Polycarpa Aurata, Spirobranchus Giganteus(sea-Christmas-tree) Table 1. These 2500 images are divided in the ratio 80: 20; 80% of data are for training purposes and 20% data are for testing.

Inclusion criteria:

- Images with higher resolution are included.
- Images with proper bounding box annotations are included.
- Images with different lighting conditions are included.
- Images from different perspective positions are included.

Exclusion criteria:

- Images with watermark and copyright text (stock photos) are excluded.
- Images without any labels are excluded. #Ambiguous features and decrease the localization confidence
- Datasets with similar kind of images are discarded. #To reduce bias and increase variance
- Discard similar kind of images from validation and train set. #Training validation overlap

Methodology:

We followed these 3 main steps in the present work shown in Figure 4:

- Image Collection/ Acquisition and Annotation
- Image enhancement and Augmentation
- Detection of Coral network architecture

Image Collection/ Acquisition and Annotations:

We collected various images of corals from the online platform, around 2500 images have been taken and they are of different dimensions. To maintain uniformity, we are using the OpenCV library to resize each image to 320 x 320 pixels. After that, for annotation purposes for labelling the images, we are using the python labelme library Figure 1: Labelme UI.

Image enhancement and Augmentations:

To train a DL model, it needs more data sets with variations like different orientation variability, contrast changes, alignment, viewing perspectives, zoom in, and zoom out. For that reason, we performed data augmentation. This approach significantly improves the visual quality of underwater images by enhancing contrast, as well as reducing noise and artifacts. In Figure 2 all augmented images are shown.

All the images were annotated by using labelme tool and generates an XML file in PascalVOC format. Annotations get divided into test_labels(20%) and train_labels(80%). By using xml_to_csv function test and train data csv files are generated for corresponding XML files, also it creates a label_map.pbtxt as shown in Figure 3:Generated label map file which contains the annotation classes in JSON format. During the study, the libraries of TensorFlow and Keras were utilized to perform coral detection and identification.

As per the process flow, all the generated CSV files were further processed to make TFRecords files which is a simple format for storing a sequence of binary records as protocol buffers. All the TFRecords formats were processed for training using SSD-MobileNet architecture. All experiment procedures were carried out on Google Collab. It uses 1xTesla k80 GPU having 2496 CUDA cores having 12 GB GDDR5 VRAM. It uses a 1x1xsingle core hyperthreaded i.e. (1 core, 2 threads) Xeon Processors @2.3Ghz CPU. It has 12.6 GB RAM.

Detection of Coral network architecture:

There are mainly three steps in an object detection framework.

1. Firstly, an algorithm or a model is employed to get regions of interest or region proposals. These region proposals are nothing but a large set of bounding boxes spanning the full image.
2. The second step consists of visual features that are extracted for each of the bounding boxes, they are further evaluated and it is determined whether and which objects present in the proposals are based on visual features (i.e., an object classification component).
3. In the third step which is the final post-processing step, overlapping boxes are combined into a single bounding box (that is, non-maximum suppression).

TensorFlow Object Detection API is used, which brings together a lot of the foregoing ideas together in a single package, allowing us to quickly repeat over different configurations using the TensorFlow backend. With the API, we define the object detection model using configuration files and therefore the TensorFlow Detection API is responsible for structuring all the necessary elements together. Figure 5: Architecture of SSD-MobileNet model describe the process, how it is working.

MobileNet

It has been found that MobileNet is best suited for underwater imaging. The MobileNet model is based on depth-separable convolutions, a form of factorized convolutions that factor a standard convolution into a depth wise convolution and a 1×1 convolution called a pointwise convolution. The standard convolution layer is parameterized by the DK× DK× M × N convolution kernel, where DK is the spatial dimension of the kernel that is supposed to be square with M as a number of input channels and N as the number of output channels. The architecture for MobileNet is defined in (Howard et al. 2017). ReLU nonlinearity

follows all layers except the final fully connected layer that has no nonlinearity and feeds into a classification SoftMax layer. Down sampling is handled in both the depth convolutions and the first layer with stride convolution. A final average pooling in front of the fully connected layer reduces the spatial resolution to 1. The architecture used in MobileNet (Howard et al. 2017) to extract image frame features is employed in this work before it is given to the network of SSD detection. The first column shows stride used by the convolving filter, with s1 being one and s2 being two steps. Corresponding second column shows the size of the filter used for convolution (dw = Depthwise). The third column of the table shows the input size in pixels for each convolution. Batch Normalization and ReLU follow all layers of convolution as shown in Figure 6: MobileNet Architecture. As the present work doesn't propose to classify the images, the last two layers which are fully connected and SoftMax layer which is basically responsible for classification purpose has been removed and SDD detection head is attached in the front.

Detection Heads:

SSD:

The researchers from Google published an SSD architecture in 2016. It presents an object detection model employing a single deep neural network combining regional proposals and feature extraction. A set of default boxes over various aspect ratios and scales is used and applied to the feature maps. The feature maps are computed by passing an image through an image classification network, thus the feature extraction for the bounding boxes is extracted in a very single step as described in Table 3. Scores are generated for every object category in every of the default bounding boxes. To better fit the bottom truth boxes, adjustment offsets are calculated for every box. Different feature maps within the convolutional network correspond with different receptive fields and are utilized to naturally handle objects at different scales. As all the computation is enclosed in a single network fairly high computational speeds are achieved. Figure 7: shows detection head working process.

Intersection over Union and Non-Maximum Suppression:

Intersection Over Union (IoU) is a number that quantifies the degree of overlap between two boxes. IoU values range from 0 to 1. Where 0 means no overlap and 1 means perfect overlap. With the help of the IoU threshold, we can decide whether the prediction is True Positive (TP), False Positive (FP), False Negative (FN). In our case we took IoU threshold **α** set at 0.5.

Non-Maximum Suppression is a computer vision method that selects a single entity out of many overlapping entities (for example bounding boxes in object detection). The criteria are usually discarding entities that are below a given probability bound. With remaining entities, we repeatedly pick the entity with the highest probability, output that as the prediction, and discard any remaining box where a IoU $\geq$ 0.5 with the box output in the previous step. Figure 8 shows the working procedure of NMS and IoU.

Results and Discussion

The proposed model was evaluated on the datasets we collected from online platforms.

We analysed the performance by mAP (mean Average Precision) and computational resource consumption. At the end of the experiment, the object detection result was used to analyse how the predicted degree of unconference affected the loss function to prove the rationality of the proposed methods.

Figure 9 shows the detection of coral reef patches from an underwater image frame. Starting from the first patch is detected with 99% accuracy. The detected coral reef from a different data set has an accuracy of 94% and 99% respectively signifying that our trained model is accurate. Also, some other detected results from the real-time video are also presented.

Results of training of Neural Network and testing of it were obtained as follows:

1. Number of epochs is 30000 and Validation Loss is 1.79.
2. Average Training accuracy of images is 100%.
3. Average Test accuracy of images is 94%.

This proposed system able to detect almost all corals whatever we have taken for our study, more efficiently with an accuracy of 94%.

We found if there are many corals present together this machine is not able to detect each and every coral perfectly with bounding boxes.

Sometimes small corals left unannotated as the image having less resolutions.

For future reference, there are many more types of corals available which are not included in this paper; so, going forward we will expand our database with better detection accuracy. Also, we will try some other detector and classifier models to find which will more suitable for our use case.

For training the convolution neural network Google Collab was used. It uses 1xTesla k80 GPU having 2496 CUDA cores having 12 GB GDDR5 VRAM. It uses a 1x1xsingle core hyper threaded i.e. (1 core, 2 threads) Xeon Processors @2.3Ghz CPU. It has a 12.6 GB RAM.

It took approx. 2 hours to train the network with 30000 steps iteration with 16 batch size.

Conclusion

The simple automatic detection technique for corals is presented in this paper. Coral reef analysis in

320×320 size image frame is performed at 2 fps from underwater environment video for each second image from the extracted frames. We implemented SSD with the MobileNet detection tracking method. In MobileNet architecture, detachable convolution is done before giving features to fully connected layers to reduce the complexity of the extraction of features. Algorithms work well for detection and tracking. A high-accuracy object detection procedure has been achieved by using the MobileNet and the SSD detector for object detection. By using SSD (single Shot Detector), we can detect objects in the fastest manner. It helps to identify individual objects in the image in the fastest way. However, SSD is also the fastest Object Detection Model in detecting smaller objects with accuracy whereas other models such as RCNN are not able to detect smaller objects accurately. Finally, we can perform Real-Time Object Detection using SSD with MobileNet which is faster and more efficient than traditional methods of object detection.

Due to the complexity in the structure of some corals and similarity between them, the prediction result we got is incorrect, however majority prediction we got are positive. In
future we are trying to increase our dataset, the increased number of dataset leads to variety of observation being made and learnt by the network. The more the data, the better statistical model will perform. The model can be trained with more types of corals in order to obtain automated tool, which can be used by marine scientist or individuals for coral annotation.

References

1. Ani Brown Mary, N, and Dharma Dejey. 2018. "Classification of Coral Reef Submarine Images and Videos Using a Novel Z with Tilted Z Local Binary Pattern (Z⊕ TZLBP)." Wireless Personal Communications 98: 2427–59.
2. Basaeed, Essa, Harish Bhaskar, and Mohammed Al-Mualla. 2016. "Supervised Remote Sensing Image Segmentation Using Boosted Convolutional Neural Networks." Knowledge-Based Systems 99 (February). https://doi.org/10.1016/j.knosys.2016.01.028.
3. Beijbom, Oscar, Peter J. Edmunds, David I. Kline, B. Greg Mitchell, and David J. Kriegman. 2012. "Automated Annotation of Coral Reef Survey Images." 2012 IEEE Conference on Computer Vision and Pattern Recognition, 1170–77.
4. Dharma, Dejey. 2018. "Coral Reef Image/Video Classification Employing Novel Octa-Angled Pattern for Triangular Sub Region and Pulse Coupled Convolutional Neural Network (PCCNN)." Multimedia Tools and Applications 77: 31545–79.
5. Elawady, Mohamed. 2015. "Sparse Coral Classification Using Deep Convolutional Neural Networks." CoRR abs/1511.09067. http://arxiv.org/abs/1511.09067.
6. Gómez-Ríos, Anabel, Siham Tabik, Julián Luengo, ASM Shihavuddin, Bartosz Krawczyk, and Francisco Herrera. 2019. "Towards Highly Accurate Coral Texture Images Classification Using Deep Convolutional Neural Networks and Data Augmentation." Expert Systems with Applications 118: 315–28.
7. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Deep Residual Learning for Image Recognition." In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–78. https://doi.org/10.1109/CVPR.2016.90.
8. Howard, Andrew G., Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." CoRR abs/1704.04861. http://arxiv.org/abs/1704.04861.

9. Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. 2017. “Densely Connected Convolutional Networks.” In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2261–69. https://doi.org/10.1109/CVPR.2017.243.
10. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. 2017. “ImageNet Classification with Deep Convolutional Neural Networks.” Commun. ACM 60 (6): 84–90. https://doi.org/10.1145/3065386.
11. Lecun, Y., L. Bottou, Y. Bengio, and P. Haffner. 1998. “Gradient-Based Learning Applied to Document Recognition.” Proceedings of the IEEE 86 (11): 2278–2324. https://doi.org/10.1109/5.726791.
12. Liu, Guojin, Zhenzhi Yin, Yunjian Jia, and Yulai Xie. 2017. “Passenger Flow Estimation Based on Convolutional Neural Network in Public Transportation System.” Knowl. Based Syst. 123: 102–15.
13. Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. 2015. “SSD: Single Shot MultiBox Detector.” CoRR abs/1512.02325. http://arxiv.org/abs/1512.02325.
14. M. Stokes, Grant Deane. 2009. “Automated Processing of Coral Reef Benthic Images.”
15. Limnology and Oceanography: Methods 7 (2): 157–68.
16. Mahmood, A, M Bennamoun, Senjian An, F Sohel, F Boussaid, R Hovey, G Kendrick, and RB Fisher. 2016. “Automatic Annotation of Coral Reefs Using Deep Learning.” In Oceans 2016 Mts/IEEE Monterey, 1–5. IEEE.
17. Mahmood, Ammar, Mohammed Bennamoun, Senjian An, Ferdous Sohel, Farid Boussaid, Renae Hovey, Gary Kendrick, and Robert B Fisher. 2017. “Deep Learning for Coral Classification.” In Handbook of Neural Computation, 383–401. Elsevier.
18. Marcos, Ma Shiela Angeli C, Maricor N Soriano, and Caesar A Saloma. 2005. “Classification of Coral Reef Images from Underwater Video Using Neural Networks.” Optics Express 13 (22): 8766–71.
19. Mary, N Ani Brown, and Dejey Dharma. 2017. “Coral Reef Image Classification Employing Improved LDP for Feature Extraction.” Journal of Visual Communication and Image Representation 49: 225–42.
20. Pizarro, Oscar, Paul Rigby, Matthew Johnson-Roberson, S. B. Williams, and Jamie Colquhoun. 2008. “Towards Image-Based Marine Habitat Classification.” OCEANS 2008, 1–7.
21. Raphael, Alina, Zvy Dubinsky, David Iluz, Jennifer Benichou, and Nathan Netanyahu. 2020. “Deep Neural Network Recognition of Shallow Water Corals in the Gulf of Eilat (Aqaba).” Scientific Reports 10 (July). https://doi.org/10.1038/s41598-020-69201-w.
22. Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, et al. 2014. “ImageNet Large Scale Visual Recognition Challenge.” CoRR abs/1409.0575. http://arxiv.org/abs/1409.0575.
23. Shihavuddin, Asm, Nuno Gracias, Rafael Garcia, Arthur Gleason, and Brooke Gintert. 2013. “Image-Based Coral Reef Classification and Thematic Mapping.” Remote Sensing, Vol. 5, Issue 4, Pp. 1809-1841 5 (April): 1809–41. https://doi.org/10.3390/rs5041809.
24. Simonyan, Karen, and Andrew Zisserman. 2014. “Very Deep Convolutional Networks for Large-Scale Image Recognition.” CoRR abs/1409.1556.
25. Song, Hong, Syed Raza Mehdi, Yangfan Zhang, Yichun Shentu, Qixin Wan, Wenxin Wang, Kazim Raza, and Hui Huang. 2021. “Development of Coral Investigation System Based on Semantic Segmentation of Single-Channel Images.” Sensors 21 (5). https://doi.org/10.3390/s21051848.

26. Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. "Rethinking the Inception Architecture for Computer Vision." In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2818–26. https://doi.org/10.1109/CVPR.2016.308.

Tables

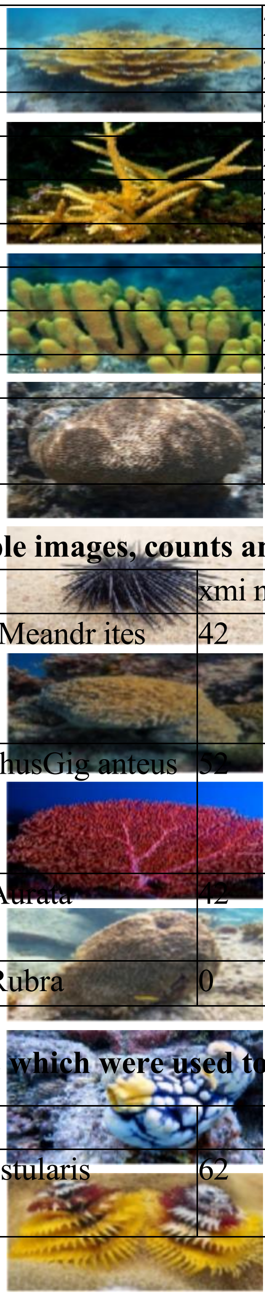
Sl. No.	Corals	Image	No. of Cor als	Identical Code
1	Acropora Palmata		250	AcroporaPalmata
2	Acropora		250	Acropora
3	Aplysina Fistularis		250	AplysinaFistularis
4	Colpophyllia Natans (Brain- Corals)		250	ColpophylliaNata ns
5	Diadema Savignyi		250	DiademaSavignyi
6	Meandrina Meandrites		250	MeandrinaMeandr ites
7	Melithaea Rubra (precious- Corals)		250	MelithaeaRubra
8	Montastraea Cavernosa		250	MontastraeaCaver nosa
9	Polycarpa Aurata		250	PolycarpaAurata
10	Spirobranchus Giganteus(sea- Christmas-tree)		250	SpirobranchusGig anteus

Table 1: Dataset Information (Sample images, counts and labelling constant)

filename	wid th	heig ht	class	xmi n	ymi n	xma x	yma x
MeandrinaMeandrites208.jpg	320	320	MeandrinaMeandr ites	42	7	282	294
SpirobranchusGiganteus101.jpg	320	320	SpirobranchusGig anteus	32	37	260	248
PolycarpaAurata29.jpg	320	320	PolycarpaAurata	42	42	214	309
MelithaeaRu	320	320	MelithaeaRubra	0	0	320	320

Table 2: A Few data from CSV Files which were used to generate TFRecords

bra13.jpg							
AplysinaFistularis18.jpg	320	320	AplysinaFistularis	62	1	187	320

DiademaSav ignyi31.jpg	320	320	DiademaSavignyi	99	30	279	283
Acroporas71 .jpg	320	320	Acroporas	33	110	295	287
AcroporaPal mata17.jpg	320	320	AcroporaPalmata	80	71	286	140
Colpophyllia Natans16.jp g	320	320	ColpophylliaNatan s	34	42	232	301

Table 3: Bounding box generation per grid in SSD Figures

Feature map sizes (Grid cells)	Bounding box per grid call	No of bounding box per Feature layer
38 x 38	4	5576
19 x 19	6	2166
10 x 10	6	600
5 x 5	6	150
3 x 3	4	36
1 x 1	4	4
Total Detection		8732

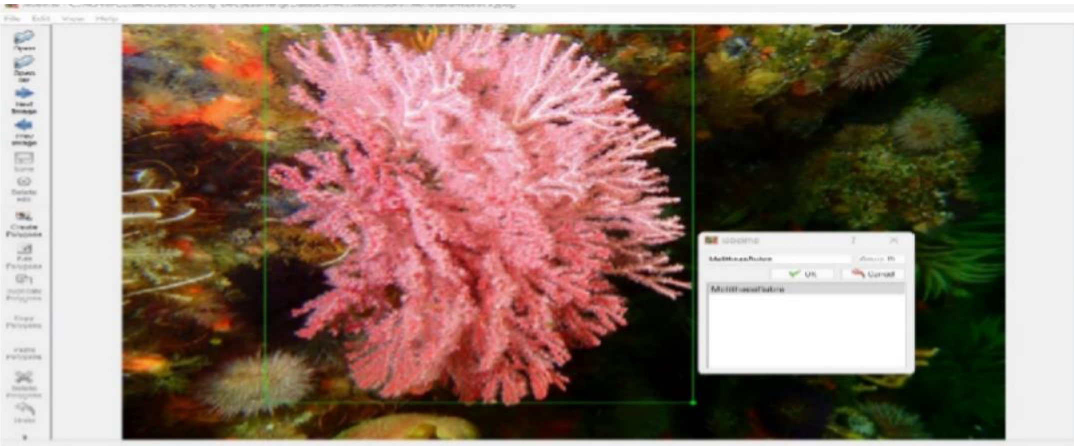
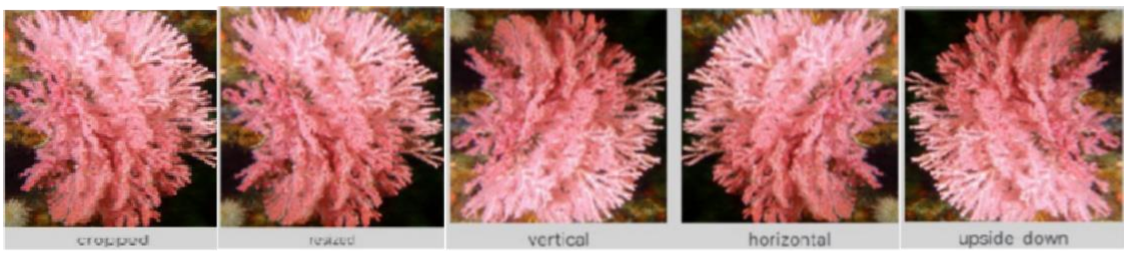
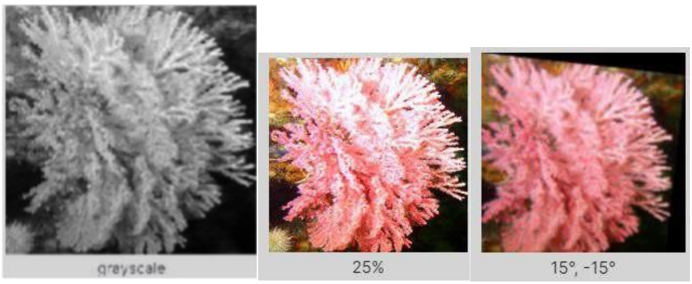


Figure 1: Labelme UI



Static crop Resized 320 x 320 Vertical Flip Horizontal Flip 90° Rotate

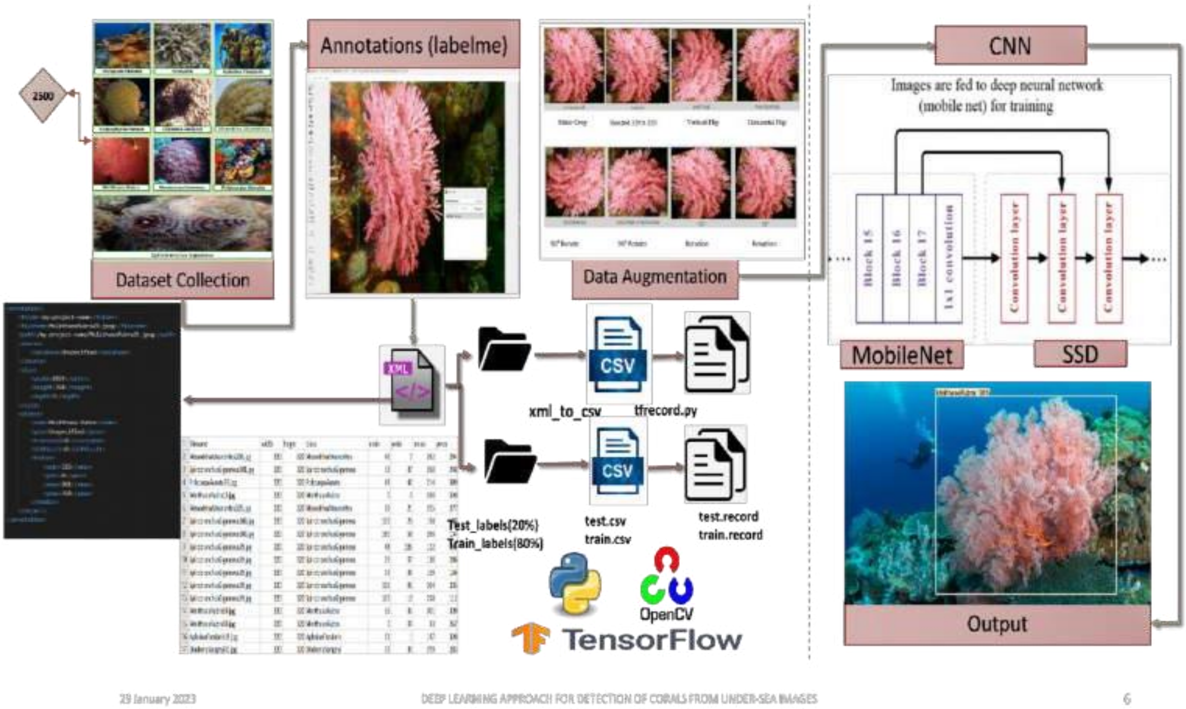


Grayscale Brightness Shear

```
item {
    id: 1
    name: 'Acropora'
}
item {
    id: 2
    name: 'AcroporaPalmata'
}
item {
    id: 3
    name: 'AplysinaFistularis'
}
item {
    id: 4
    name: 'ColpophylliaNatans'
}
item {
    id: 5
    name: 'DiademaSavignyi'
}
item {
    id: 6
    name: 'MeandrinaMeandrites'
}
item {
    id: 7
    name: 'MelithaeaRubra'
}
item {
    id: 8
    name: 'MontastraeaCavernosa'
}
item {
    id: 9
    name: 'PolycarpaAurata'
}
item {
    id: 10
    name: 'SpirobranchusGiganteus'
}
```

Figure 2: Augmented Image

Figure 3: Generated label map



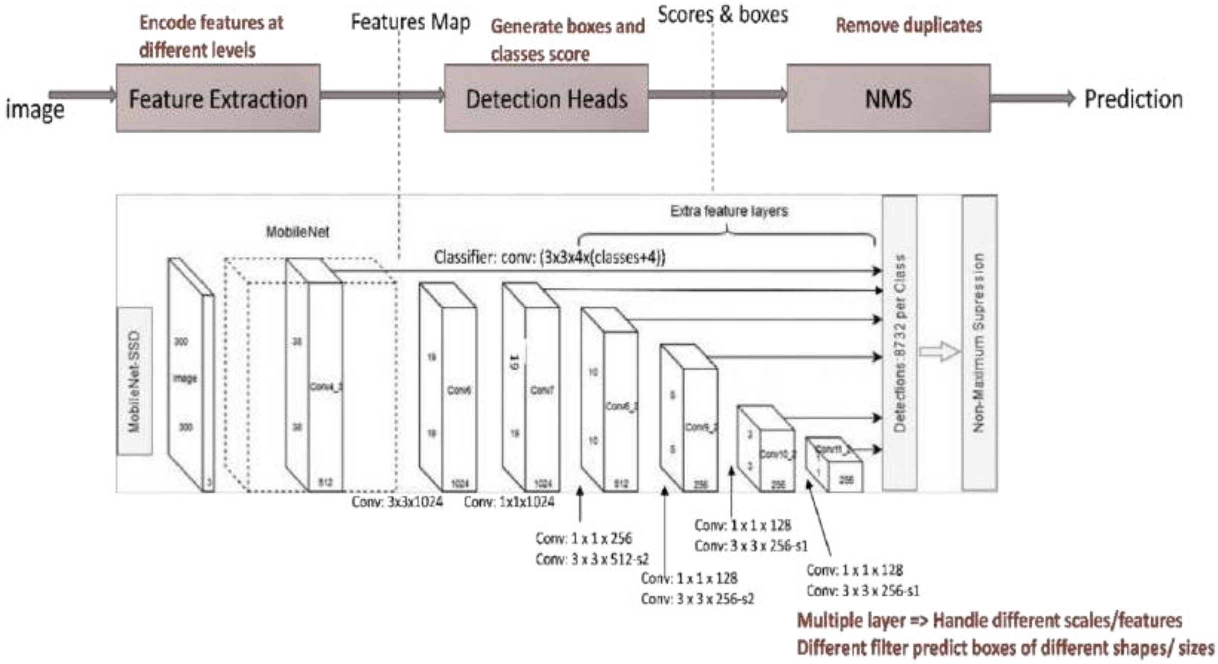


Figure 4: Methodology

Figure 5: Architecture of SSD-MobileNet model

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
$5\times$ Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

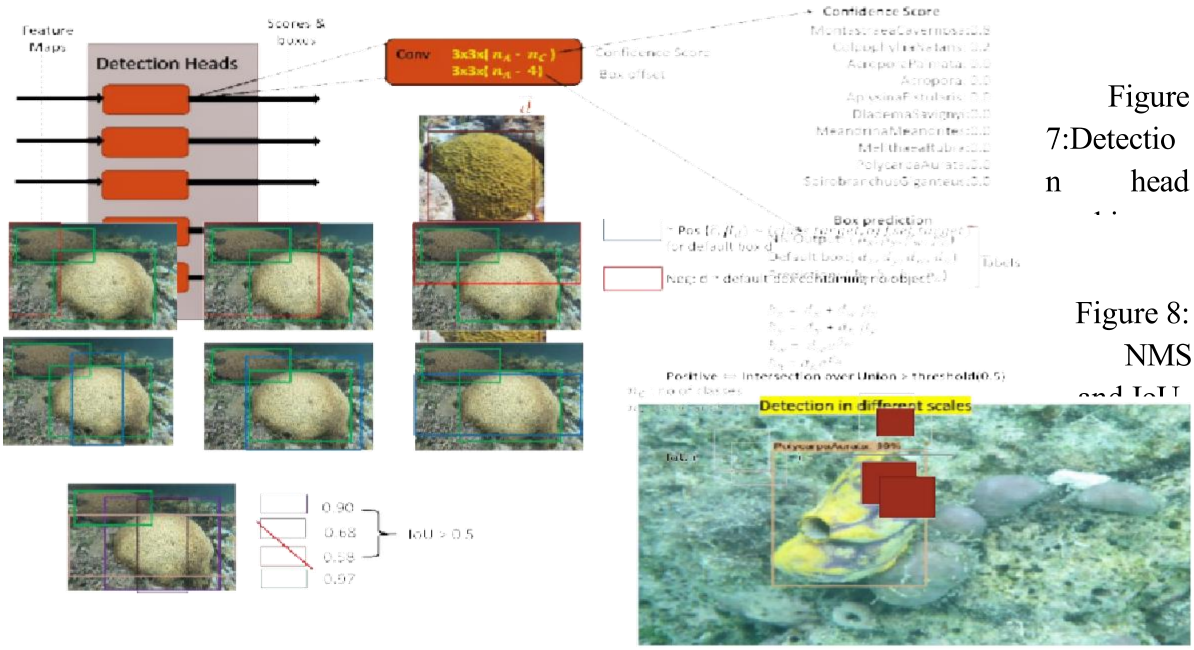


Figure 7:Detectio n head

Figure 8: NMS and IoU

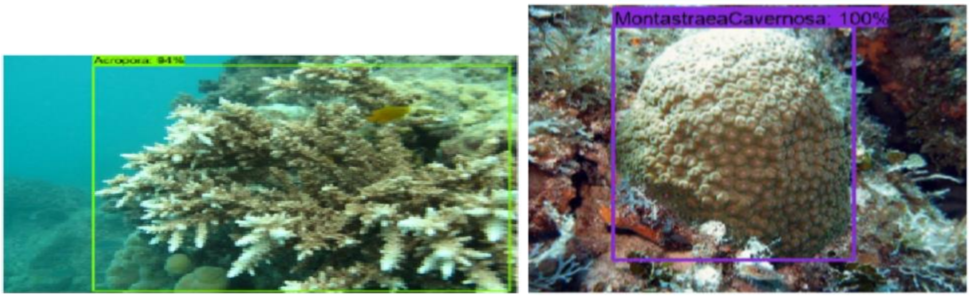


Figure 6: MobileNet Architectur

Figure 9: Results obtained from Image and Real-time video