# Great Barrier Reef COTS Detection

Aashika Padmanabhan, Nikita Pardeshi,
Mitali Shah, Shazia Sulaiman, Rae Zhang

# Agenda

01

## INTRODUCTION

A brief description of the problem

02

## DATASET

Dataset description and formatting

03

## MODELS

- Faster RCNN
- YOLOv5

04

## FUTURE SCOPE & CONCLUSION

What improvements can be made?

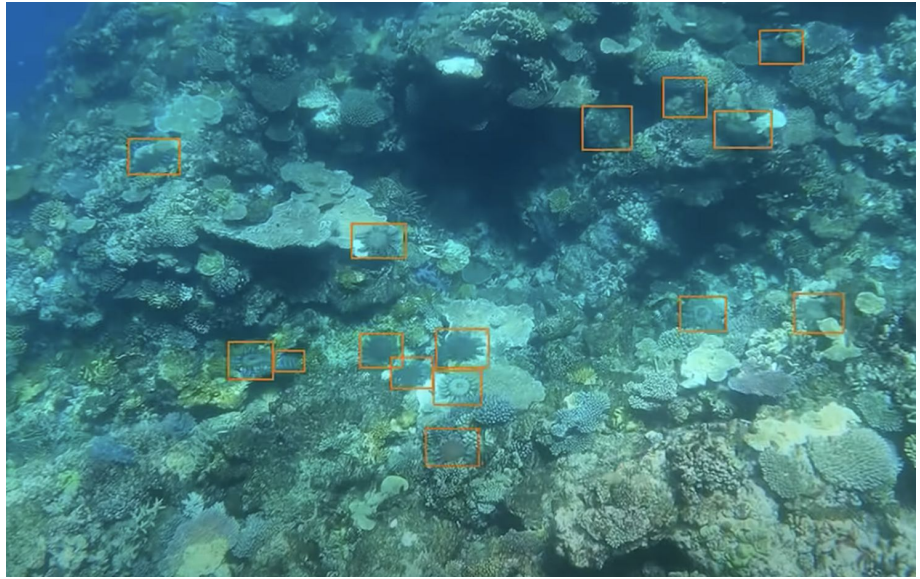
# Introduction [1]

- Great Barrier Reef is a complex of coral reefs, shoals, and islets in the Pacific Ocean. UNESCO issued a second recommendation to include this site in the Danger list.
- COTs are marine invertebrates that feed on coral and can reach plague proportions and devastate hard coral communities(coral bleaching)
- Since 1962, crown-of-thorns starfish outbreaks have had a major impact on the many reefs. During an outbreak, which is when 15 or more COTS are found in a one-hectare area, the starfish can strip a reef of 90% of its living coral tissue.
- COTS are covered in spikes containing toxins and eat by extruding their stomachs out from their bodies, wrapping it around corals and digesting their tissues.

Image Source

# Data Information

| video_id | sequence | video_frame | sequence_frame | image_id | annotations |
|---|---|---|---|---|---|
| 0 | 40258 | 13 | 13 | 0-13 | [] |
| 0 | 40258 | 14 | 14 | 0-14 | [] |
| 0 | 40258 | 15 | 15 | 0-15 | [] |
| 0 | 40258 | 16 | 16 | 0-16 | [{'x': 559, 'y': 213, 'width': 50, 'height': 32}] |
| 0 | 40258 | 17 | 17 | 0-17 | [{'x': 558, 'y': 213, 'width': 50, 'height': 32}] |
| 0 | 40258 | 18 | 18 | 0-18 | [{'x': 557, 'y': 213, 'width': 50, 'height': 32}] |
| 0 | 40258 | 19 | 19 | 0-19 | [{'x': 556, 'y': 214, 'width': 50, 'height': 32}] |
| 0 | 40258 | 20 | 20 | 0-20 | [{'x': 555, 'y': 214, 'width': 50, 'height': 32}] |
| 0 | 40258 | 21 | 21 | 0-21 | [{'x': 550, 'y': 214, 'width': 50, 'height': 32}] |



- train_images
  - video_0 - 6,709 images
  - video_1 - 8,233 images
  - video_2 - 8,562 images

TOTAL - 23,504 images

# Faster RCNN Model

# RCNN, Fast RCNN[2,3,8]

- Problems with RCNN
  - Takes lot of time to classify 2000 regions per image
  - Selective Search is a fixed algorithm with no learning
- Problems with Fast RCNN
  - Region proposal slows down the process

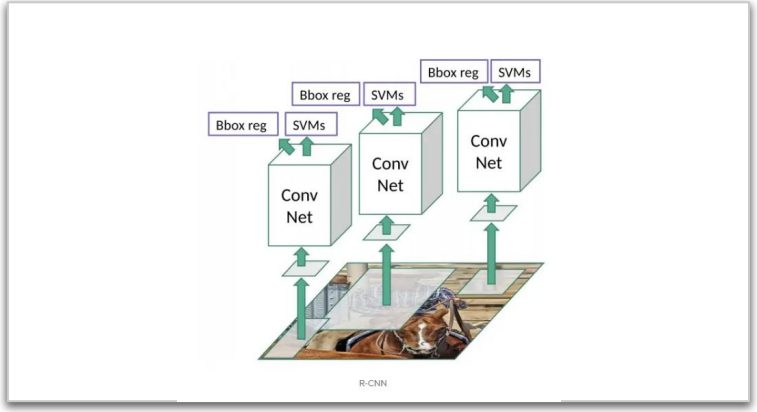



Figure 1: Source

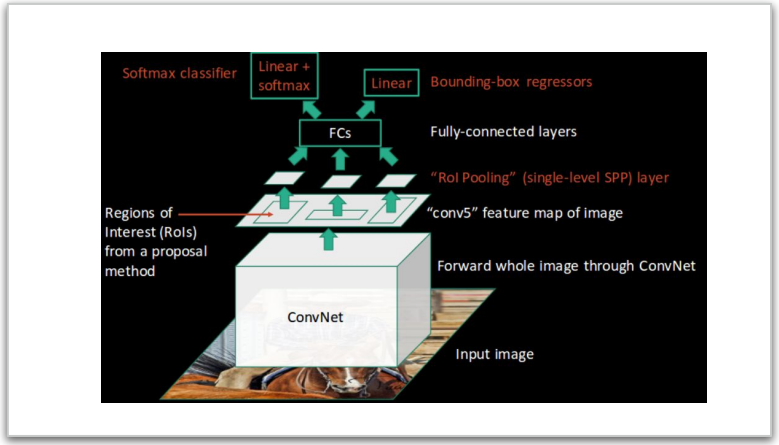



Figure 2: Source

# Faster RCNN Model[2,5]

R-CNN & Fast R-CNN uses selective search to find out the region proposals - slow and time-consuming.

Faster RCNN eliminates the selective search algorithm and let the network learn the region proposals.

Faster RCNN is composed from 3 parts:

- Part 1 : Convolution layers
- Part 2 : Region Proposal Network (RPN)
- Part 3 : Classes and Bounding Boxes prediction

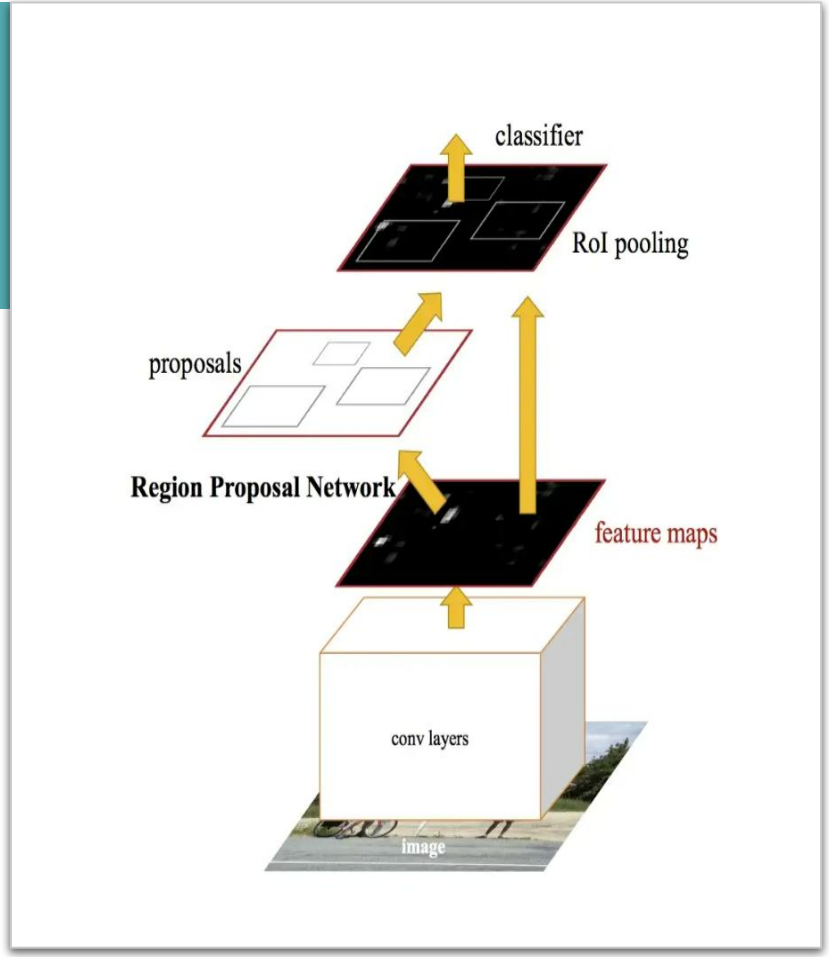
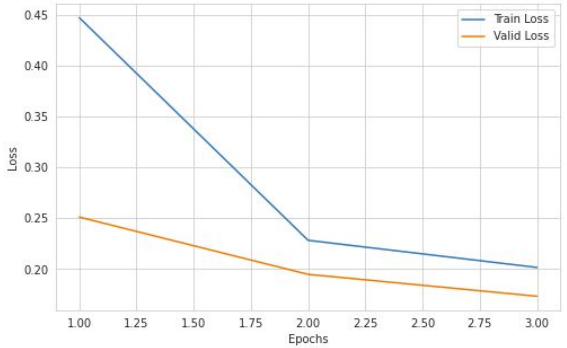
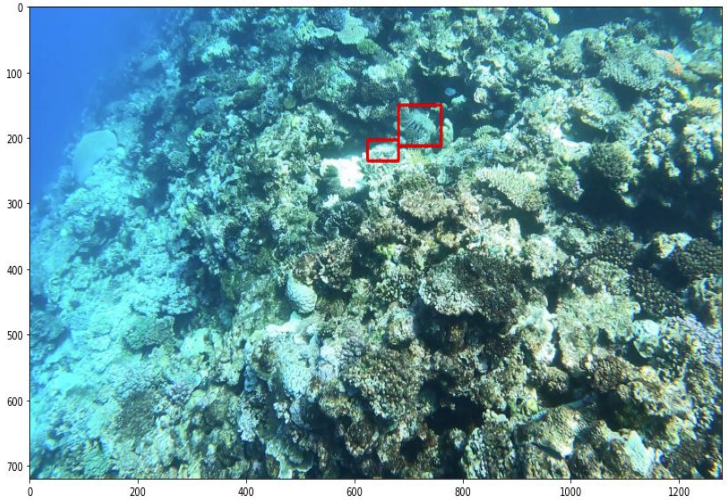


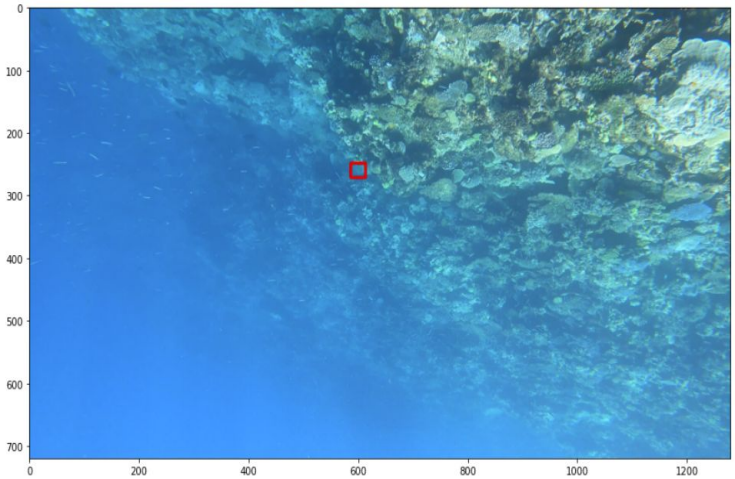

Figure 3: Source

# Faster RCNN Model Results

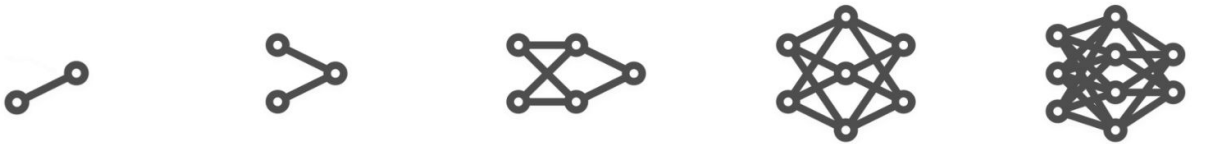
Performed on a subset of data.



# Fine Tune YOLOv5s



| Nano | Small | Medium | Large | XLarge |
|---|---|---|---|---|
| YOLOv5n | YOLOv5s | YOLOv5m | YOLOv5l | YOLOv5x |
| 4 $MB_{FP16}$<br>6.3 $ms_{V100}$<br>28.4 $mAP_{COCO}$ | 14 $MB_{FP16}$<br>6.4 $ms_{V100}$<br>37.2 $mAP_{COCO}$ | 41 $MB_{FP16}$<br>8.2 $ms_{V100}$<br>45.2 $mAP_{COCO}$ | 89 $MB_{FP16}$<br>10.1 $ms_{V100}$<br>48.8 $mAP_{COCO}$ | 166 $MB_{FP16}$<br>12.1 $ms_{V100}$<br>50.7 $mAP_{COCO}$ |

# YOLOv5 [7, 9, 10, 11]

- YOLOv5 launched in 2020, is one of the most advanced object identification algorithms available.
- YOLOv5 comes in four main versions: small (s), medium (m), large (l), and extra large (x), each offering progressively higher accuracy rates. Each variant also takes a different amount of time to train.
- Uses CNN that detects objects using a single neural network to process the entire picture and predicts bounding boxes and probabilities of each component.

**Architecture:**

- **Backbone:** used to extract key features/ useful characteristics from an input image

- **Neck:** used to create feature pyramids. Feature pyramids help models in object scaling by identifying the same object in various sizes and scales.

- **Head:** final detection where anchor boxes are used to construct final output vectors with class probabilities, object scores and bounding boxes.

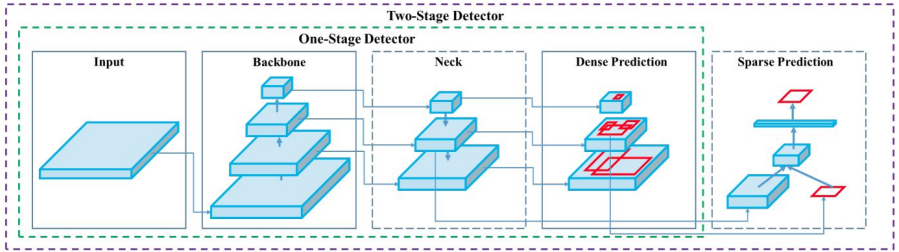



Figure 4: Source

# Data Workflow

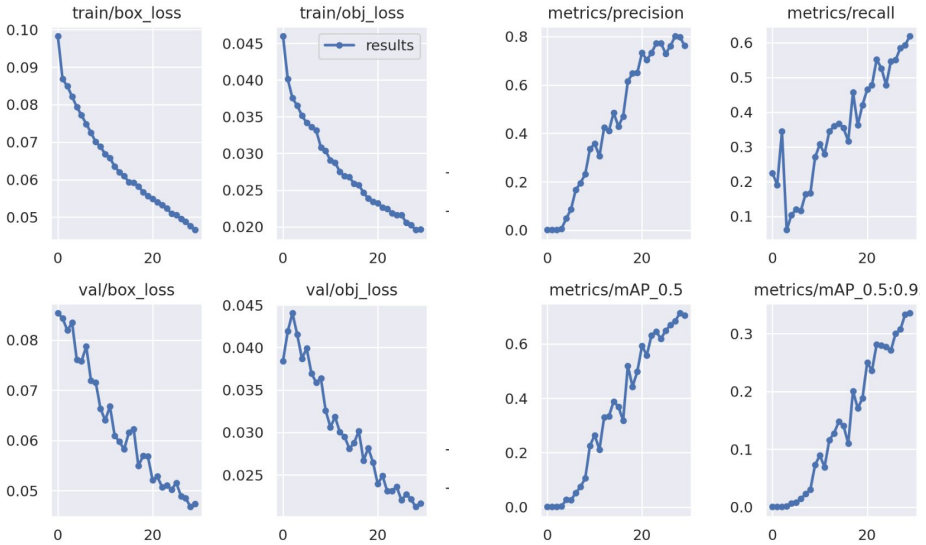
| | video_id | sequence | video_frame | sequence_frame | image_id | annotations | no_of_bboxes |
|---|---|---|---|---|---|---|---|
| **0** | 0 | 40258 | 16 | 16 | 0-16 | [{'x': 559, 'y': 213, 'width': 50, 'height': 32}] | 1 |
| **1** | 0 | 40258 | 17 | 17 | 0-17 | [{'x': 558, 'y': 213, 'width': 50, 'height': 32}] | 1 |
| **2** | 0 | 40258 | 18 | 18 | 0-18 | [{'x': 557, 'y': 213, 'width': 50, 'height': 32}] | 1 |

| | video_id | sequence | video_frame | sequence_frame | image_id | annotations | no_of_bboxes | bbox | image_path |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 40258 | 16 | 16 | 0-16 | [{'x': 559, 'y': 213, 'width': 50, 'height': 32}] | 1 | [[0, 0.45625, 0.7986111111111112, 0.0390625, 0... | /content/drive/.shortcut-targets-by-id/1BJ54wg... |
| **1** | 0 | 40258 | 17 | 17 | 0-17 | [{'x': 558, 'y': 213, 'width': 50, 'height': 32}] | 1 | [[0, 0.45546875, 0.7972222222222223, 0.0390625... | /content/drive/.shortcut-targets-by-id/1BJ54wg... |
| **2** | 0 | 40258 | 18 | 18 | 0-18 | [{'x': 557, 'y': 213, 'width': 50, 'height': 32}] | 1 | [[0, 0.4546875, 0.7958333333333333, 0.0390625,... | /content/drive/.shortcut-targets-by-id/1BJ54wg... |

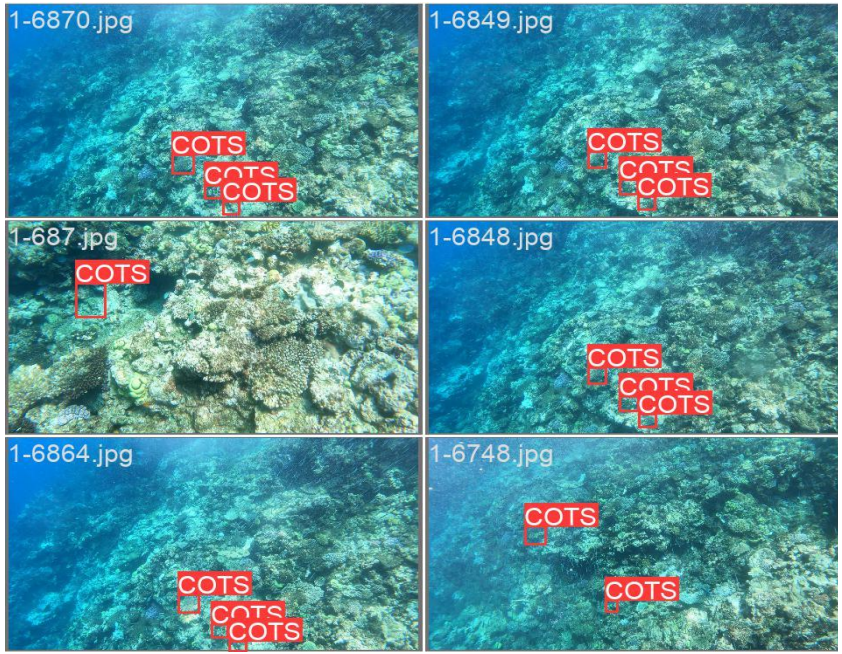
# Training & Validation [6, 9]

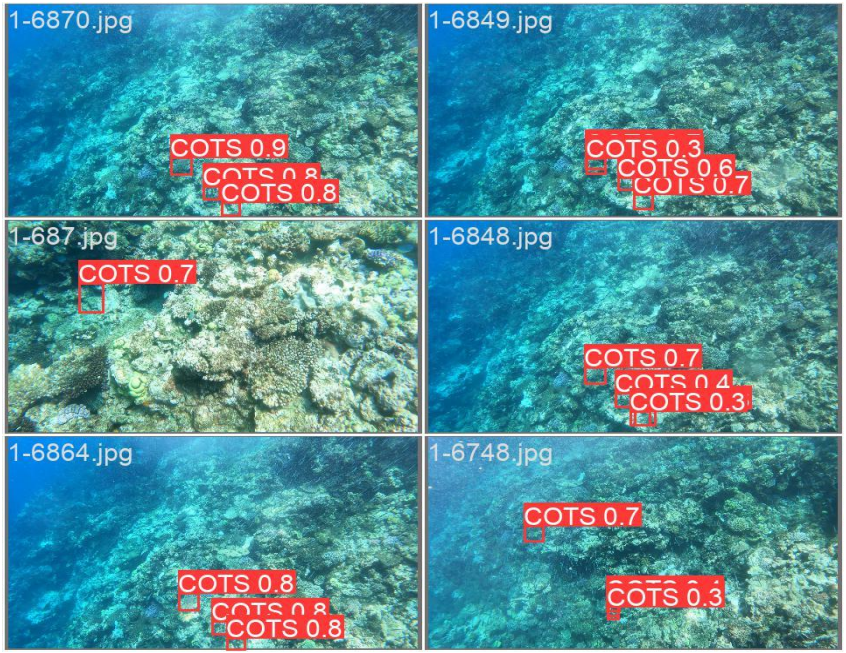
- The dataset was trained on 30 epochs and the weights were saved as per pytorch model format (.pt).
- Trained on Google Colab GPU.

Evaluation metrics: There are various metrics used to measure the accuracy and performance of an object detection model – precision, recall, and mAP (mean average precision)

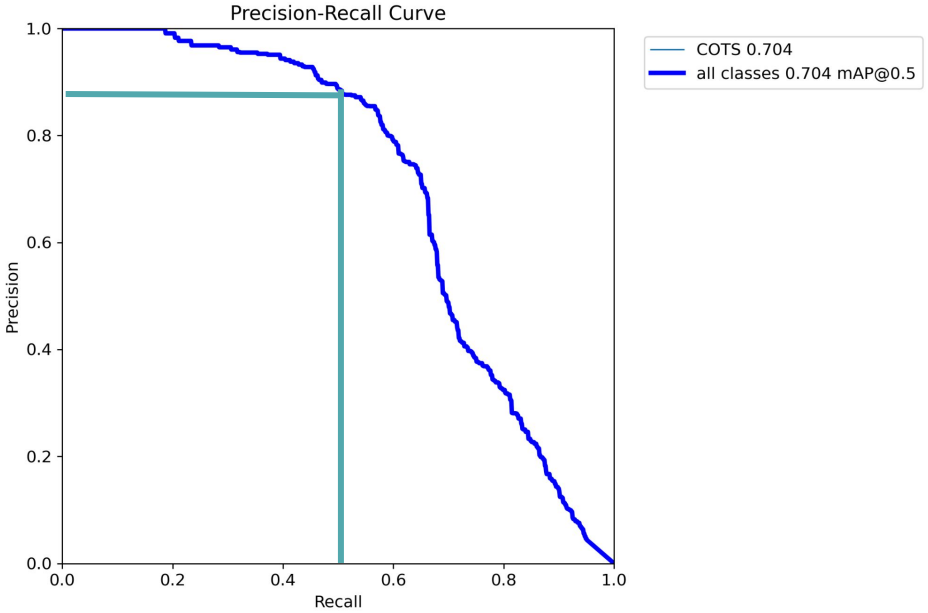


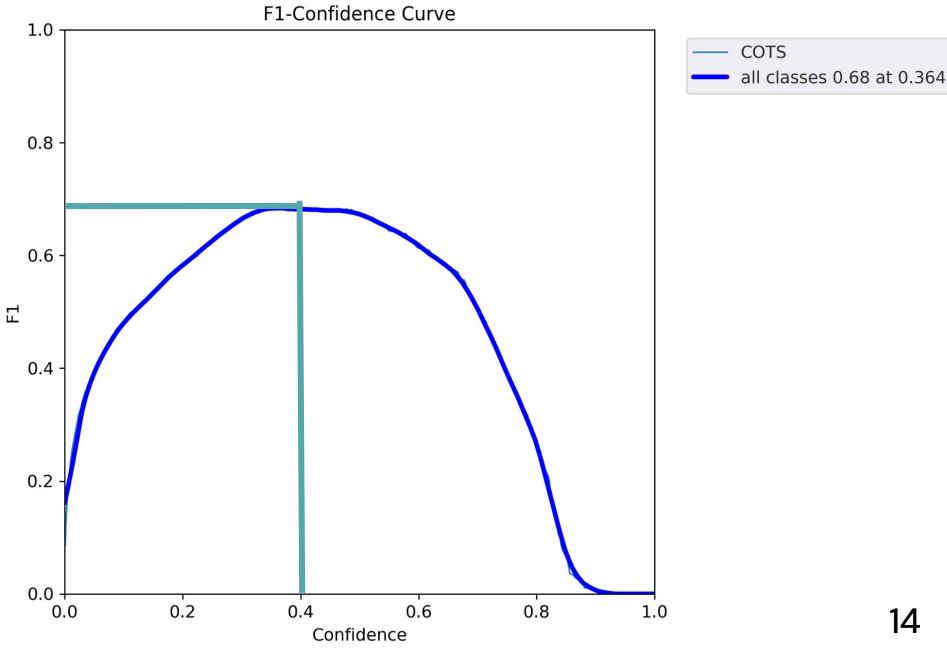
# YOLOv5 Results



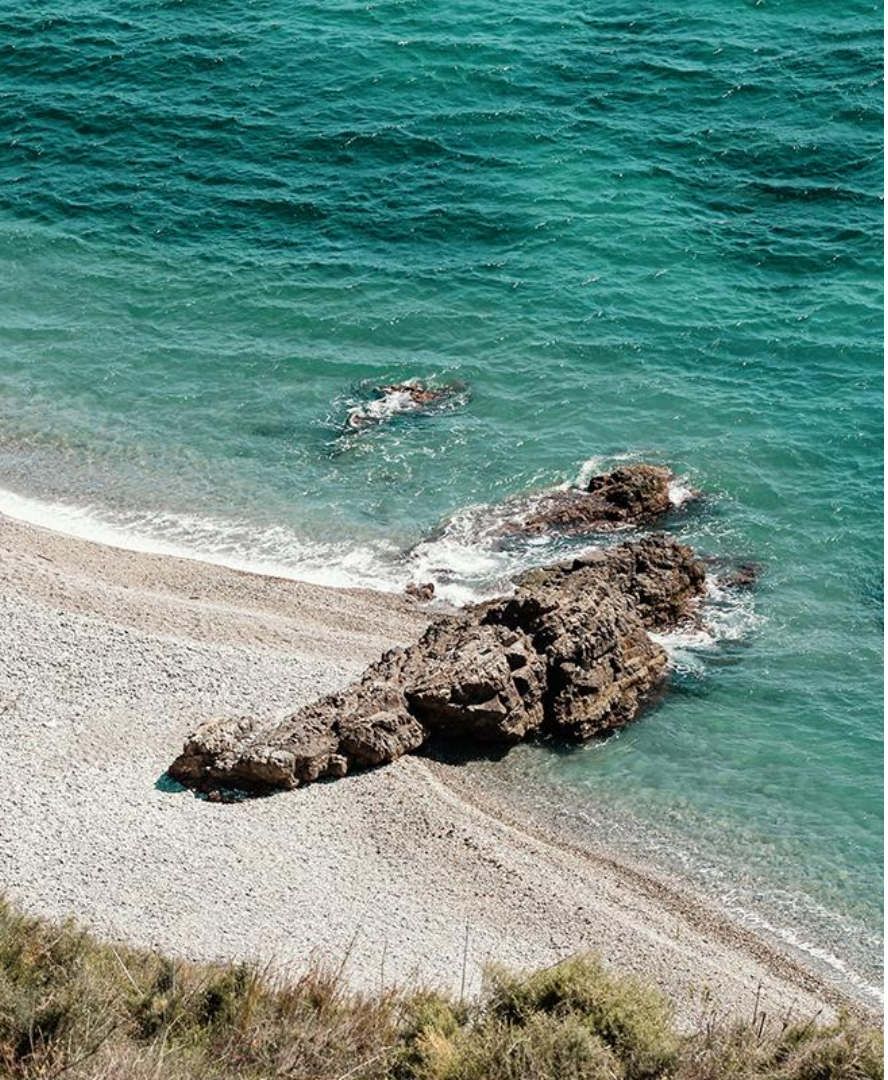
# YOLOv5 Evaluation



# Future Scope & Conclusion [9]

- YOLOv5 is much faster than Faster R-CNN model.
- YOLO is capable of processing the image only once and simultaneously detecting and classifying the images.
- The accuracy of YOLOv5 model was better than Faster R-CNN.
- We only trained YOLOv5 with 30 epochs; increasing the epochs could give better results.
- YOLOv5s has very less parameters compared to YOLOv5m, YOLOv5l, YOLOv5x and YOLOv5x6. Hence, these models can be explored too with higher computation power.
- Ensemble techniques using YOLOv5 and YOLOX (another version) can be explored.



# Thank you!

# References


1. *Crown-of-thorns starfish control*. Great Barrier Reef Foundation. (n.d.). Retrieved December 6, 2022, from https://www.barrierreef.org/what-we-do/reef-trust-partnership/crown-of-thorns-starfish-control
2. Gandhi, R. (2018, December 3). *R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms*. Medium. https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e
3. DeepLearning. (2020, December 4). *R-CNN, Fast R-CNN and Faster R-CNN explained*. YouTube. https://www.youtube.com/watch?v=vr5rs_cTKCs
4. *Crown-of-thorns starfish - reef resilience*. (n.d.). Retrieved December 6, 2022, from https://www.reefresilience.org/pdf/COTS_Nov2003.pdf
5. you359. (n.d.). *You359/Keras-FASTERRCNN: Keras implementation of faster R-CNN*. GitHub. Retrieved December 6, 2022, from https://github.com/you359/Keras-FasterRCNN
6. Solawetz, J. (2022, November 21). What is YOLOv5? A Guide for Beginners. Roboflow Blog. https://blog.roboflow.com/yolov5-improvements-and-evaluation/
7. Garg, A. (2021, December 14). How to Use Yolo v5 Object Detection Algorithm for Custom Object Detection. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/12/how-to-use-yolo-v5-object-detection-algorithm-for-custom-object-detection-an-example-use-case/
8. Sharma, P. (2020, May 24). *A Step-by-Step Introduction to the Basic Object Detection Algorithms (Part 1)*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2018/10/a-step-by-step-introduction-to-the-basic-object-detection-algorithms-part-1/
9. Tips for Best Training Results 📌 - YOLOv5 Documentation. (n.d.). https://docs.ultralytics.com/tutorials/training-tips-best-results/
10. Mean Average Precision (mAP) in Object Detection. Kukil. https://learnopencv.com/mean-average-precision-map-object-detection-model-evaluation-metric/
11. Bochkovskiy, A. (2020, April 23). YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv.org. https://arxiv.org/abs/2004.10934
12. Maunil2k/The-Great-Barrier-Reef-Kaggle: YOLOv5 trained on custom dataset for COTS detection - Kaggle competition. Maunil2k. https://github.com/Maunil2k/The-Great-Barrier-Reef-Kaggle