Detecting Crown of Thorns Starfish using Yolov5

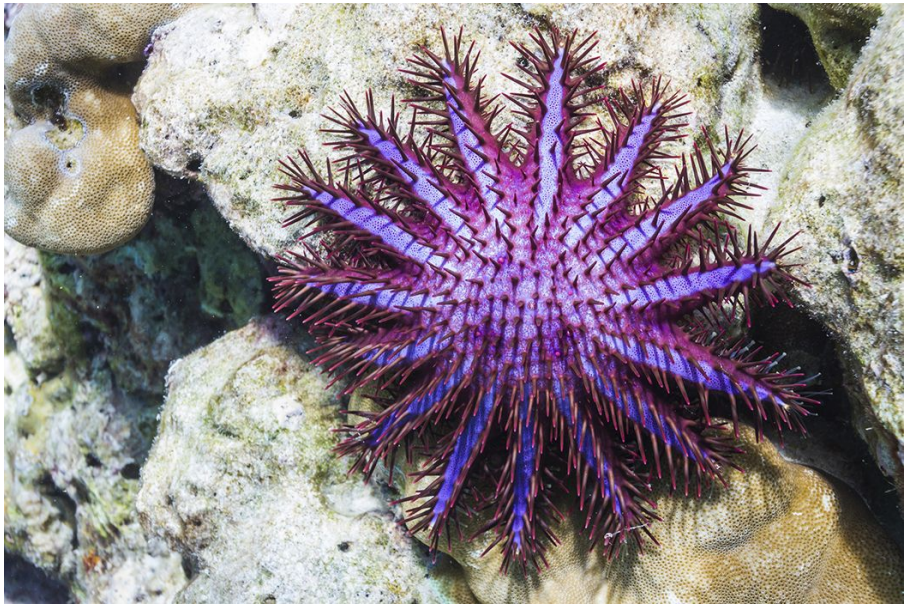
Ian Boulis and Alan Bouwman

Background

Coral Reefs are one of seven wonders of the natural world and thousands of fish and underwater creatures call this place their home

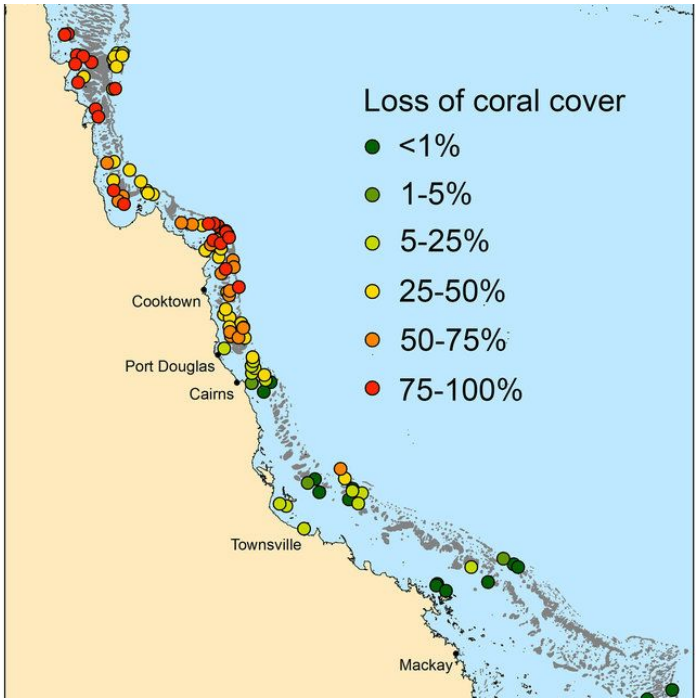
Currently under threat of overpopulation from the Crown of Thorns Starfish (COTS)

Great Barrier Reef Foundation, Australia's Natural Science Agency, Google, and Kaggle have teamed up to apply machine learning to this overpopulation problem.



Source: https://daily.jstor.org/when-crown-of-thorns-starfish-attack/

Motivation



Source:https://www.researchgate.net/figure/Loss-of-coral-cover-along-the-Great-Barrier-Reef-in-2016-Losses-measured-on-110-reefs_fig2_324595286

The Great Barrier Reef is one of seven wonders of the natural world and deserves to be protected and preserved for as long as possible.

The reef has recently seen a massive loss of coral and fish population, and this project aims to prevent further loss from the overpopulation of COTS.

Dataset Description

Data consists of videos taken of the seafloor by a diver being towed underwater.

Videos are taken frame by frame and any COTS in frame are labeled with bounding boxes.

Original dataset contains several folders and files, however, we will only be using the folder of images and folder of labels in our experiment.

The folder of images consists of every frame of the three videos taken.

The folder of labels consists of text files corresponding to every frame of the video.

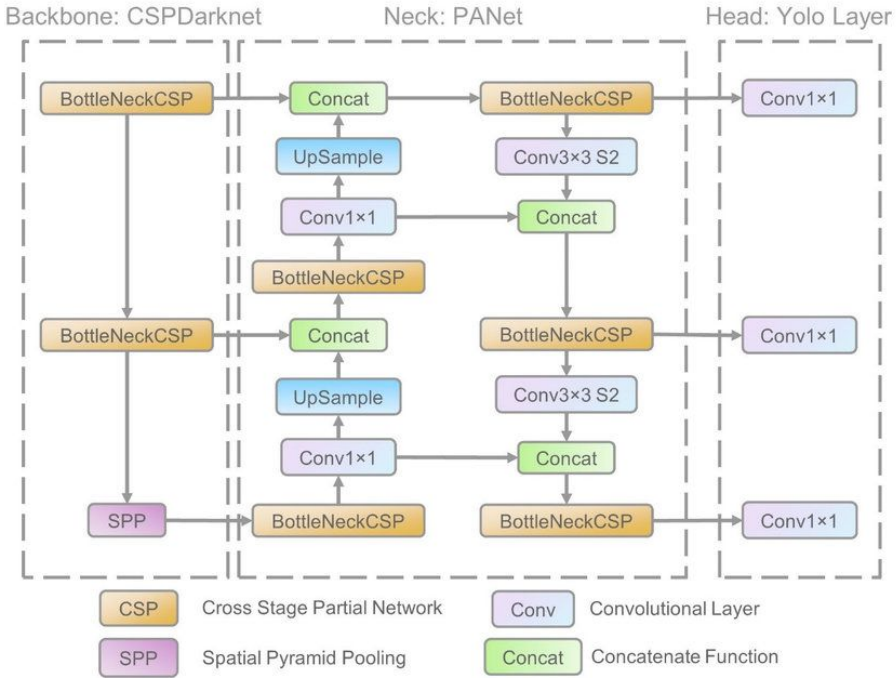
- These text files contain the bounding box annotations for any COTS in the given frame.
  - Each box has a class label, the x and y coordinates of the box center, and height and length of each box.

Our original dataset consists of 23501 images.

Architecture

Yolov5 architecture consists of several different frameworks and can be broken down into three phases: model backbone, model neck, and model head.

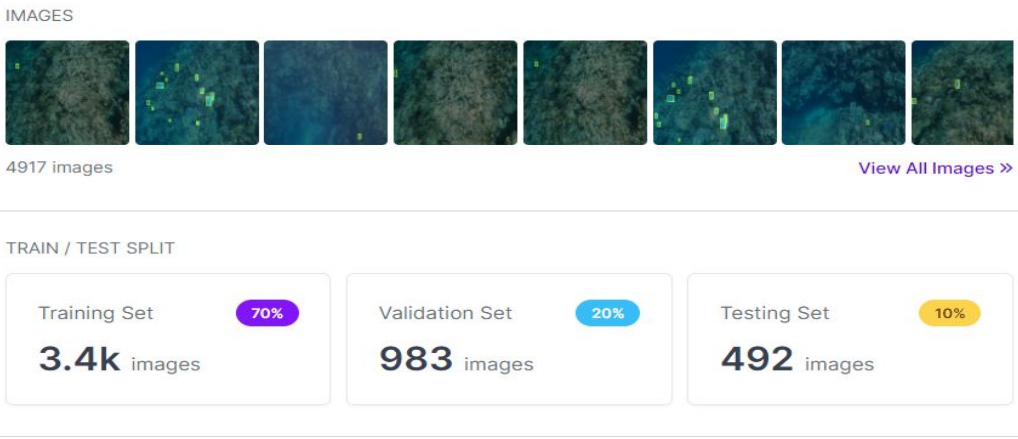
1. Model Backbone
    a. Used to extract image features
    b. Implements Cross Stage Partial networks
2. Model Neck
    a. Generates feature pyramids and generalizes features
    b. Implements Path Aggregation Networks
3. Model Head
    a. Used to make predictions and calculate confidence scores, box parameters, and accuracy.
    b. Consists of several CNNs



Source: https://www.researchgate.net/figure/The-network-architecture-of-Yolov5-It-consists-of-three-parts-1-Backbone-CSPDarknet_fig1_349299852

Data Preprocessing

Utilized Roboflow to access our data.
Roboflow automatically annotated images
With a free account, number of images is capped
Used their API to access data within Google Colab

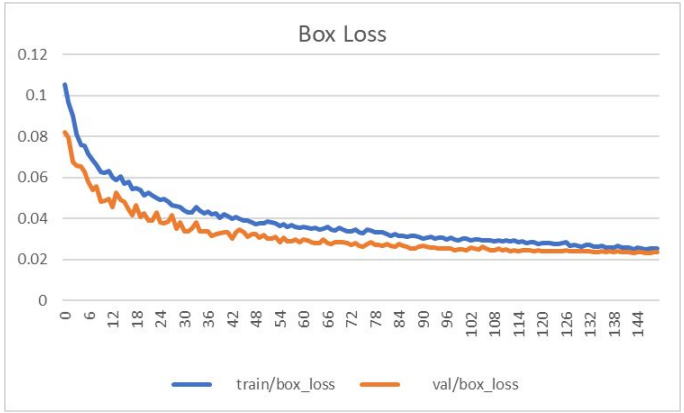


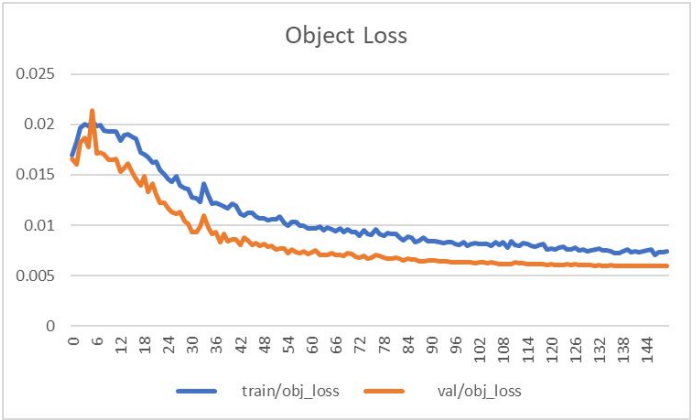
Roboflow automatically applies some preprocessing steps for us.

1. Auto-orientate
2. Auto-scale
3. Splits dataset

We are capped at 4917 images, with 3400 in the training set, 983 in validation, and 492 in testing set

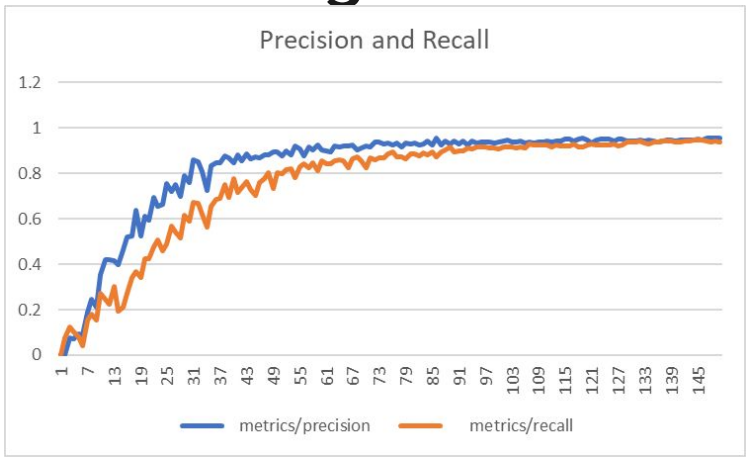
Training

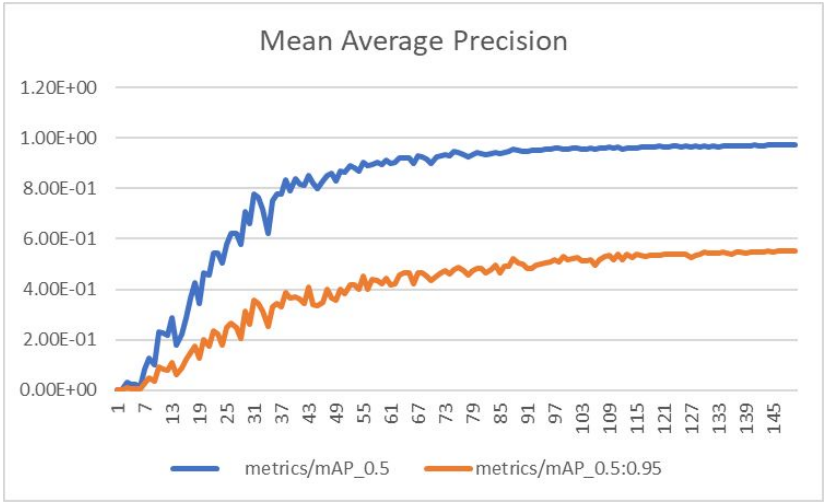




Both plots indicate the model converges on the testing and validation data. The models may be slightly over fit, especially for box loss.

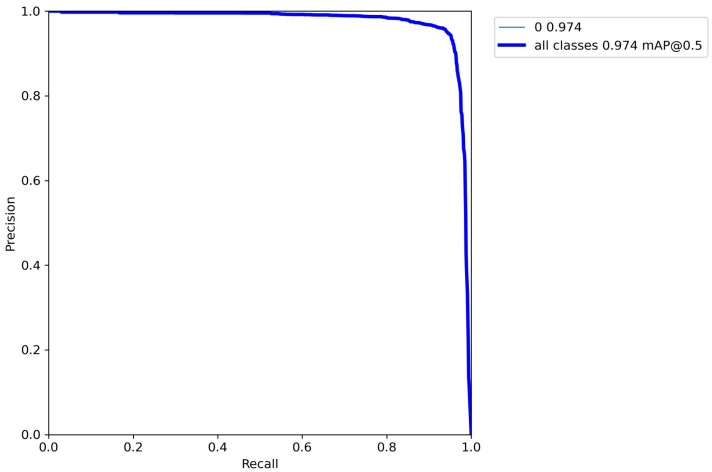
Training Cont.

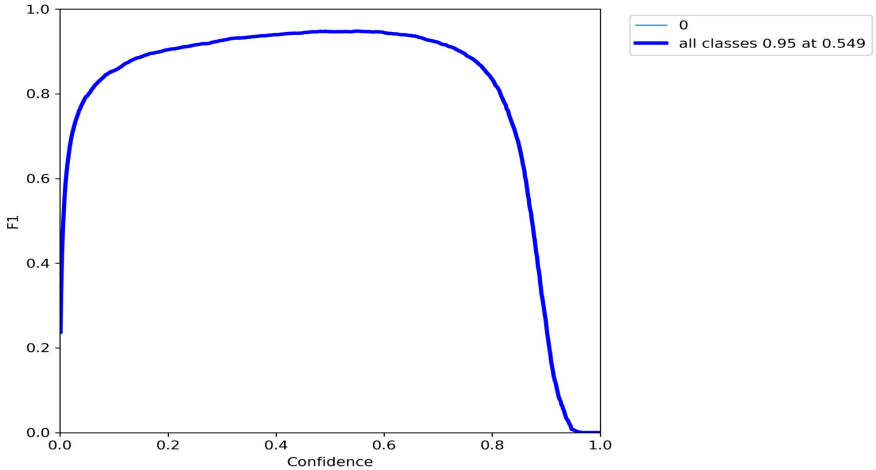




Both plots indicate the model performs well on the validation data and convergers as epochs continue. The mAP[0.5:0.05:0.95] is not very high (only about 0.58) meaning that the IoU at a high threshold does not perform as well.

Experimental Results





AUC of PR curve = 0.947. F1 Score is maximized with confidence threshold of 0.549. Both curves indicated model is performing well.

Conclusion

Limitations:

- No evaluation on testing data
  - Yolov5 does not provide ways to assess the model on the test data with significantly alteringing source code
- No Object Detection
  - Object Detection Could have been overkill
- No Comparison with Other Models

Success:

- Model performs well enough to detect invasive COTS starfish in the Great Barrier Reef
- Very good performance metrics on training and validation data
- Learned a lot about the theory of object detection and gained practical experience with implementation and APIs