Custom Object detection using Machine Learning on Sagemaker

Introduction

Shipping traffic is growing fast. More ships increase the chances of infractions at sea like environmentally devastating ship accidents, piracy, illegal fishing, drug trafficking, and illegal cargo movement. This has compelled many organizations, from environmental protection agencies to insurance companies and national government authorities, to have a closer watch over the open seas. Airbus offers comprehensive maritime monitoring services by building a meaningful solution for wide coverage, fine details, intensive monitoring, premium reactivity and interpretation response. Combining its proprietary-data with highly-trained analysts, they help to support the maritime industry to increase knowledge, anticipate threats, trigger alerts, and improve efficiency at sea.

Airbus has recruited your team to locate ships in satellite images, and put an aligned bounding box segment around the ships you locate. A long term contract will be provided to your team based on a Proof-of-Concept model that you provide to Airbus. Airbus has been doing most of their analytics and machine learning on-prem, in their own data centers, and is interested in learning about the advantages of using AWS Sagemaker for model training and hosting. Currently, Airbus is producing more satellite data than it can process, and is interested in an automated solution.

About the Dataset

This dataset is focused on recognition of ships by machine learning. This dataset contains a database of small images of tankers, commercial ships or fishing ships. It also contains some large scale SPOT images at 1.5 m. resolution to test your ship detection algorithm on real satellite images.

The data can be downloaded here:

https://www.kaggle.com/c/airbus-ship-detection/data

(**Hint**: *Also check out recent submissions/ discussions)*

The size of this dataset is 3.8 GB. It contains 7,074 images (256px by 256px) with a portion of a ship on it. The objects.csv contains a list of images with ships on it indicating the file path, the class (cls), the top left corner (tlx, tly) and the bottom right (xbr, ybr) of the ship. It also contains 20,000 more images containing background (i.e. images with no ships).

Many images do not contain ships, and those that do may contain multiple ships. Ships within and across images may differ in size (sometimes significantly) and be located in open sea, at docks, marinas, etc.

The `train_ship_segmentations.csv` file provides the ground truth (in run-length encoding format) for the training images. The `sample_submission` files contains the images in the `test` images.

Airbus data scientists realized that thinking of the problem as having two distinct steps helped:

1. Remove noise from the original image and only consider a “mask” or minimal representation of the image. This “mask” (center image below) is what they were predicting as an output using ML
2. Draw a bounding box around the predicted mask

Image Classification vs. Image Segmentation

Using the first image on the left as an example, a classification model predicts *whether there is a ship (or a car, or a bike, or a ...)* in the image; a segmentation model predicts *where an object is* in an image (the “mask”). In other words, we are labeling whether a pixel belongs to a ship (white colored in the mask image), or not (black in the mask image). While there are purely image processing techniques to create masks and segment images, these are limited to simple images with distinct features like contrasting colors (for e.g., this would work if all ships were bright red, the water was bright blue, and clouds never formed on earth). Your task is to train a model to *learn* how to segment an image. Consider the following example image:

Humans in the image are “segmented” from vehicles, signs, the road, buildings and the sky, and these different objects are colored differently as a post-processing step. Therefore, apart from recognizing different objects, we also have to delineate the boundaries of each object. Therefore, unlike classification, we need dense pixel-wise predictions from our models.

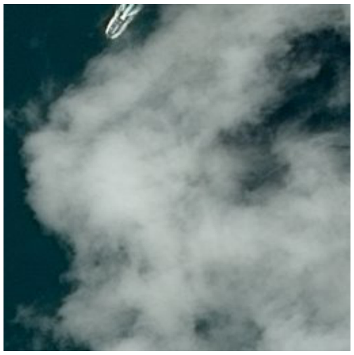
Our ship example sounds a lot simpler now, doesn't it?

Additional notes from Airbus regarding the dataset

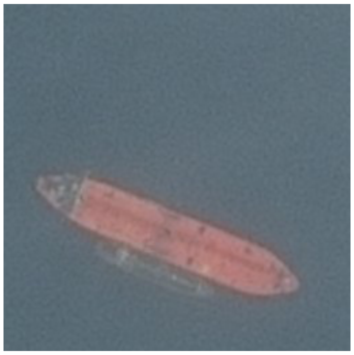
WHAT YOU ARE UP AGAINST...

Your objective is to assess the generalization capabilities of an ML model throughout the very diverse images that may be acquired. To that extent, you will assess the model performance in varied circumstances with an emphasis on difficult cases such as the following situations:

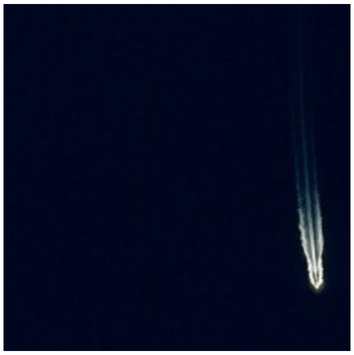
- Presence of clouds near the ships



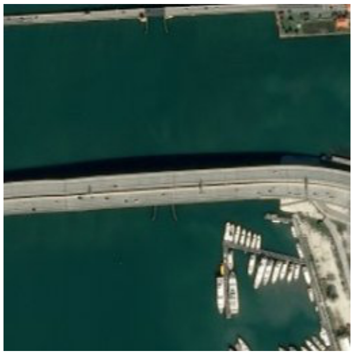
- Presence of ships under haze or thin clouds



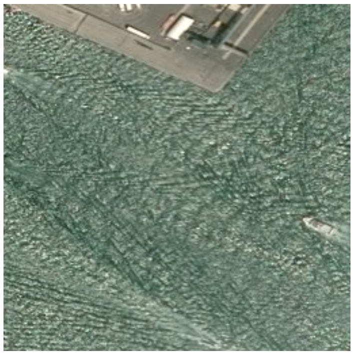
- Presence of large wake behind the ship (that will not count as part of the ship)

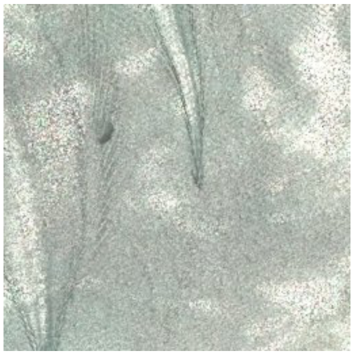


- Identification of individuals ships in marinas (Need to reduce false alarm rate on coastal areas as well as docks)

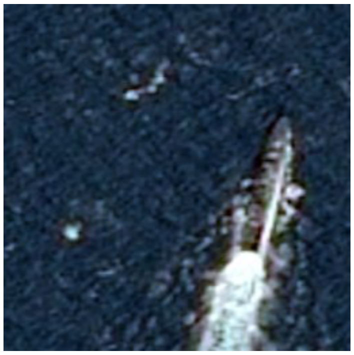


- Difficult cases due to sun reflectance on the sea





- confusion with waves due to wind conditions



- ships only partially visible in the image (in which cases the bounding box might be truncated). Ships size will greatly vary (from very small leisure ships to large tankers)

- Individual ships discrimination when two or more ships are coupled together

- Ships discrimination from other floating objects such as buoys, barges, wind turbines, etc

REFERENCES

https://www.itm-conferences.org/articles/itmconf/pdf/2017/04/itmconf_ita2017_05012.pdf

https://www.kaggle.com/c/airbus-ship-detection/data

https://doi.org/10.1117/1.JRS.11.042611

https://arxiv.org/pdf/1411.4038.pdf

https://thehonestanalytics.com/automatic-identification-systems-market-2018-global-industry-analysis-by-key-players-segmentation-trends-and-forecast-by-2023/