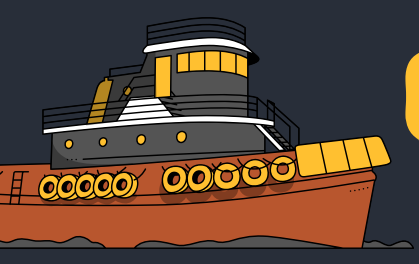


OBJECT DETECTION
SHIP AERIAL IMAGES

PROBLEM OVERVIEW

The primary objective of our project is to develop a deep learning based object detection model specifically designed to identify ships in aerial images in a satellite point of view. The object detection model that we developed later can be use to detect ships that pass through a certain area of our location accurately.

GROUP 3

- Aaron Winston Gho - 2702210522
- Fidelis Yoga Pradipta - 2702296224
- Kylee Valencia - 2440052583
- Matthew Anthony Tjahjadi 2702404576
- Matthew Farrell Kurniawan 2702280850
- Rowan Walyata Alfajri 2702295663

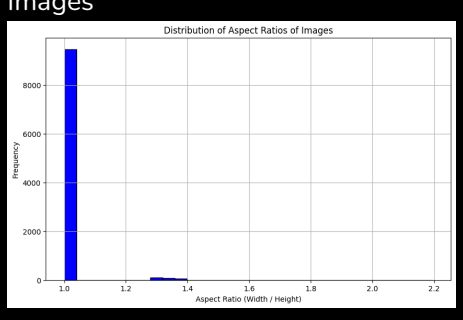
DATA PREVIEW

- The dataset was obtained from Kaggle with 13,435 images in it.
- Our dataset is split into train, test, and validation, containing 9,697 images for training, 1,573 for testing, and 2,165 for validation.

SAMPLE VISUALIZATION IMAGE

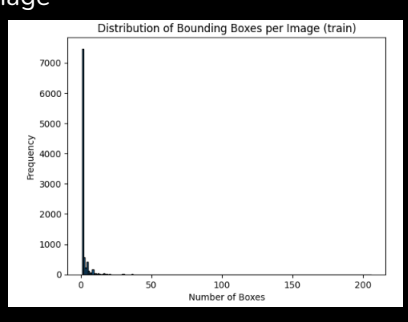
EXPLORATORY DATA ANALYSIS

Distribution of Aspect Ratios of Images



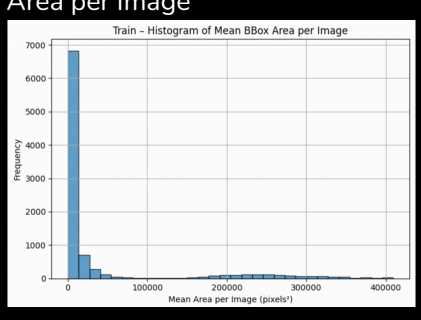
- Most images have a 1:1 aspect ratio, but some deviate.
- Consistent aspect ratio helps prevent distortion and misaligned bounding boxes.
- This ensures better model accuracy during training and inference.

Distribution of Bounding Box per Image



- Most images have 1–5 boxes, some have many, causing a right skewed distribution.
- Images with zero boxes will be labels as background in testing.
- Outlier images will be split or limited.
- All images will be resized to a fixed size (512 x 512).

Histogram of Mean Bounding Box Area per Image



- Bounding box sizes vary due to different ships and zoom levels.
- Many are too small (low pixel count).

We'll drop them because:

- Lack visual details.
- Hard for model to learn from.
- Prone to noise, unreliable labels.
- Skew training distribution.

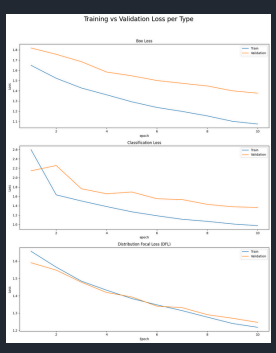
DATA PREPROCESSING

1. Dropping low bounding box pixel size
2. Dropping high frequency count bounding box
3. Resizing ratio to 1:1 for bounding box and image
4. Resizing height & weight for image, scale for the bounding box following the change for the height & weight

MODEL

- In this project, we implemented three object detection models: YOLOv8s, YOLOv8m, and YOLOv11.
- For each model, we trained it for 10 epochs with a batch size of 4.

MODEL 1 : YOLOV8S



Final Loss Values at Epoch 10
Box Loss :
-Train: 1.0745,
-Validation: 1.3764

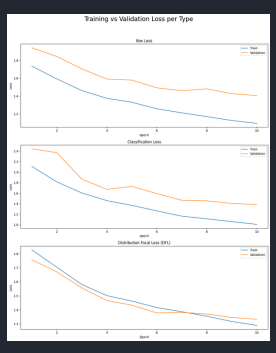
Classification Loss :
-Train: 0.9759
-Validation: 1.3654

DFL Loss :
-Train: 1.2175
-Validation: 1.2459

From the loss curves of both train and val, we can see that they both decrease over epochs. however the graphs havent plateued, meaning there could potentially be improvements if we were to increase the epochs for training in the future

yolov8s metric reuslts:
Precision: 0.771
Recall: 0.365
mAP50: 0.587
maAP50-95: 0.448

MODEL 2 : YOLOV8M



Final Loss Values at Epoch 10
Box Loss :
-Train: 1.0933
-Validation: 1.4057

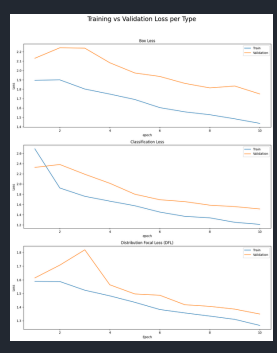
Classification Loss :
-Train: 1.0068,
-Validation: 1.3836

DFL Loss :
-Train: 1.2885,
-Validation: 1.3321

From the loss curves of both train and val, we can see that they both decrease over epochs. however the graphs havent plateued, meaning there could potentially be improvements if we were to increase the epochs for training in the future

yolov8m metric results:
Precision: 0.729
Recall: 0.375
mAP50: 0.573
maAP50-95: 0.424

MODEL 3 : YOLOV11



Final Loss Values at Epoch 10
Box Loss :
-Train: 1.4359,
-Validation: 1.7496

Classification Loss :
-Train: 1.2101,
Validation: 1.5122

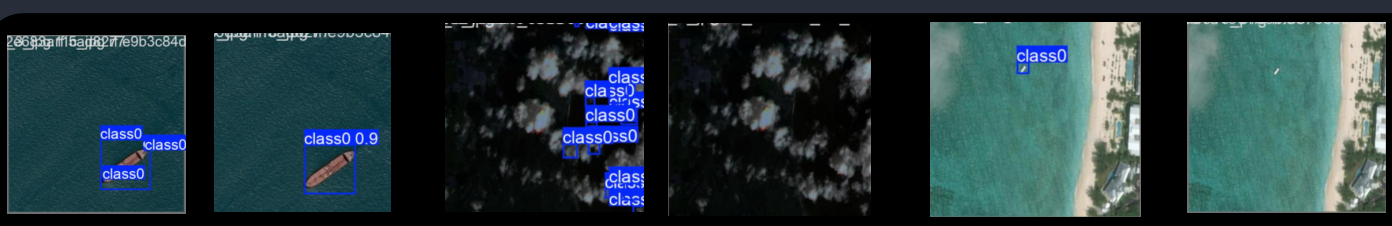
DFL Loss :
-Train: 1.2664,
-Validation: 1.3496

From the loss curves of both train and val, we can see that they both decrease over epochs. however the graphs havent plateued, meaning there could potentially be improvements if we were to increase the epochs for training in the future

yolov11 metric results:
Precision: 0.7388973966309341
Recall: 0.2594086021505376
mAP50: 0.5175344509913872
maAP50-95: 0.3517684404478509

All models performed very similarly in terms of metric results, where each metric experiences similar regardless of the model used. Although YOLOv8s is the smaller model, it performed better than YOLOv8m in terms of precision and mAP on our dataset. This was unexpected, as we initially assumed the larger and more complex model would yield better results. One possible explanation is that our dataset, around 9,000 training images with only one class, is not complex enough to benefit from a larger model. Another factor could be that YOLOv8m wasn't trained long enough to fully showcase its capabilities. As a result, we will conduct testing using YOLOv8s.

TESTING: YOLOV8S



CONCLUSION

From running the test data through our Yolov8s model, we can see a few examples of detection, where our model can successfully detect ships where they are very obviously visible. However in cases where the ships are small and rather hard to see, our model fails to detect so. There are also cases where the raw data is wrongly labelled, however our model can correctly predict the classes of the wrongly labelled.