# Detecting sea vessels in SAR imagery using Deep Learning

Syed Riaz Raza
syedriaz.raza@studenti.unipd.it

Syed Fahad Hassan
syedfahad.hassan@studenti.unipd.it

Mutahir Yaseen
mutahir.yaseen@studenti.unipd.it

## Abstract

*This project focuses on developing a deep learning solution for detecting small vessels in Synthetic Aperture Radar (SAR) imagery. By training a specialized convolutional neural network (CNN) on a synthetic dataset, we aim to improve accuracy and efficiency compared to traditional methods. The outcomes will enhance maritime surveillance systems, aiding in vessel detection for safety and security purposes.*

## 1. Introduction

### 1.1. Description

SAR provides high-resolution, all-day, all-weather satellite imagery, which has become one of the most important means for high-resolution ocean observation and is well suited to better understanding the maritime domain. We build and train a computer vision small vessel detection model that automatically generates bounding boxes around maritime vessels using the open source SAR dataset LS-SSDD-v1.0 [10]. This type of automation would enable regulatory agencies to perform shipwreck rescue, fishery enforcement, and vessel traffic management more effectively. Our implementation is open source and available on Github[1].

### 1.2. Chalenges

Detecting objects in SAR imagery is difficult for a variety of reasons. Some ship and non-ship targets, such as waves, dams, islands, icebergs, or reefs, have approximate backscattering intensity in SAR imagery, making ship detection difficult [1]. Furthermore, many low- and mid-level image features that have been widely used in object detection and classification applications cannot be directly introduced into ship detection using SAR imagery, posing an additional challenge. Finally, small object detection in large-scale remote sensing images remains an unsolved problem in the computer vision literature.

[1]https://github.com/SRiazRaza/Vision-and-cognitive-systems

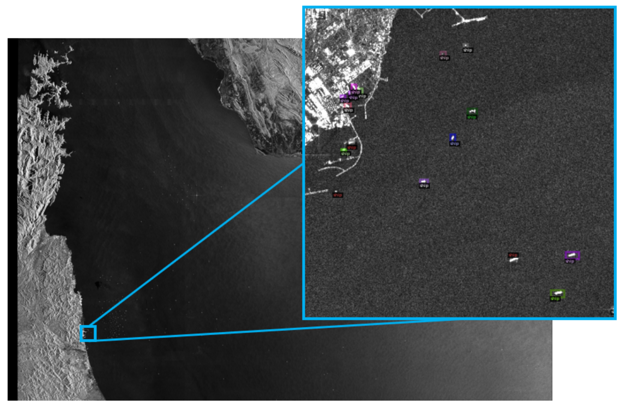

Figure 1. An example of large-scale image and sub-image from LS-SSDD-v1.0 SAR Imagery Dataset.

### 1.3. Related Work

[4, 3] provide a survey of works related to deep learning-based object detection, specifically small object detection. We use the dataset and benchmark results provided by [10] for this project, which has related research in [12, 9, 13]. Data Augmentation approaches for small object detection can be found in [5, 6].

## 2. Dataset

We make use of Sentinel-1's Large-Scale SAR Ship Detection Dataset-v1.0 (LS-SSDD-v1.0) [10]. 15 large-scale, 24,000 x 16,000 pixel SAR images are included in LS-SSDD-v1.0. As seen in Figure 1, the original large-scale SAR images are divided into 9000, 800800 pixel sub-images. The first 10 of the 15 large-scale images are used as the training set by the authors (train). The test set consists of the last five large-scale images (test). 2234 offshore sub-images (test offshore) and 766 inshore sub-images are further divided into test (test inshore). Table 1 and Figure 2 show the following dataset differences:

| Dataset | Imgs | Ships | %Imgs w/Ships | Ship /*Img |
|---|---|---|---|---|
| train | 6000 | 3637 | 0.18 | 3.23 |
| test | 3000 | 2378 | 0.24 | 3.23 |
| test_off | 2234 | 1495 | 0.27 | 2.41 |
| test_in | 766 | 883 | 0.15 | 7.54 |

Table 1. LS-SSDD-v1.0 datasets. *Img denotes an image that has at least one ship. test_off means test offshore and test_in means test inshore

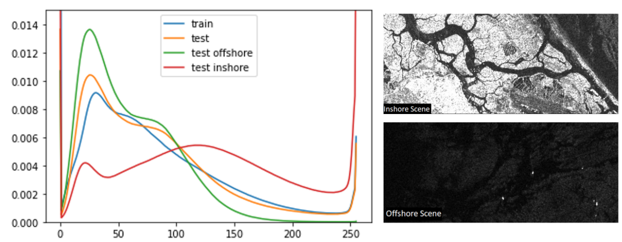


Figure 2. (Left) Sample of pixel intensity frequencies. (Right) Inshore and Offshore scenes.

### 2.1. Evaluation Matrices

We apply the intersection over union (IOU) threshold of 0.5, the non-maximum suppression (NMS) threshold of 0.5, and the score threshold of 0.5 for vessel detection. The following evaluation metrics are taken into account during the test in order to verify the baseline model to [10]: Detection Probability (Pd), False Alarm (Pf), Missed Detection (Pm), Recall, Precision, Mean Average Precision (mAP), and F1 score. The results of the baseline model are compared to those of additional modelings using only mAP as the evaluation metric.

## 3. Learning Methodology

### 3.1. Baseline Models

We implemented two fundamental models from Facebook's Detectron2 API [11, 7] to simulate the results stated in [10]. We used two neural network architectures that were both pre-trained on the COCO dataset: (1) Faster Region-Based Convolutional Neural Network (Faster R-CNN) [**?**] and (2) RetinaNet [8]. We used ResNet-50 with Feature Pyramid Network (FPN) pre-trained on the ImageNet dataset as the backbone for both of these models [2]. To guarantee adherence to references

[10] Using mini-batch gradient descent (MGD) with momentum for 12 epochs (36k iterations) and matching hyperparameters when possible given different APIs used, we trained on the entire train (no validation dataset). Table 2 compares the outcomes of the Original Paper (OP) and Baseline Model (BM).

A significantly lower level of performance was seen when

| Dataset (Source) | Model | mAP |
|---|---|---|
| test (OP) | Faster R-CNN | 74.80 |
| test offshore (OP) | Faster R-CNN | 89.99 |
| test inshore (OP) | Faster R-CNN | 46.76 |
| test (OP) | RetinaNet | 54.31 |
| test offshore (OP) | RetinaNet | 76.15 |
| test inshore (OP) | RetinaNet | 17.29 |
| test (BM) | Faster R-CNN | 71.32 |
| test offshore (BM) | Faster R-CNN | 89.00 |
| test inshore (BM) | Faster R-CNN | 38.64 |
| test (BM) | RetinaNet | 53.02 |
| test offshore (BM) | RetinaNet | 73.44 |
| test inshore (BM) | RetinaNet | 16.89 |

Table 2. Comparison between Original Paper (OP) and Baseline Model (BM) results.

inference was done while traveling, as shown in Table 3. As stated in [10], we also noticed that test offshore consistently performs much better than test inshore when inference was conducted on both tests, which is most likely because there is no near-shore backscattering. We have reason to believe that train and test come from different distributions and that train's distribution may be more similar to test inshore's distribution (referring back to Table 1 and Figure 2). In order to more accurately estimate out-of-sample performance on train, a new approach must be introduced. The Faster R-CNN baseline will be referred to as Baseline for the remainder of this essay.

| Dataset | @36k | @72k | @108k | @144k |
|---|---|---|---|---|
| train | 52.32 | 55.02 | 62.83 | 65.81 |
| test | 71.32 | 70.90 | 70.40 | 67.96 |

Table 3. Baseline performance on train versus test at various training iterations. All @iterations with mAP

### 3.2. *Baseline* to *Improved*: Validation Set, Learning Rate Scheduling, and ResNext

We depart from [10] and use train' and validation datasets on an improved model called Improved in order to more accurately estimate out-of-sample performance. Following a random train shuffle, we divided the remaining 15% of the train into validation and the remaining 85% into train'. In order to compare the improved results to the baseline and [10] test, nothing has changed. We can schedule learning rates based on validation mAP, which is another advantage of using validation. In particular, if the current mAP score is within a relative threshold of 0.01 of the best mAP observed with one epoch patience, we scale the learning rate by 0.1. To address the issue shown in Table 3, we fit a Faster R-CNN with a ResNeXt-101-32x8d FPN backbone to benefit from its state of the art results [15]. A schematic of ResNeXt block architecture compared to ResNet block is shown in Figure3.

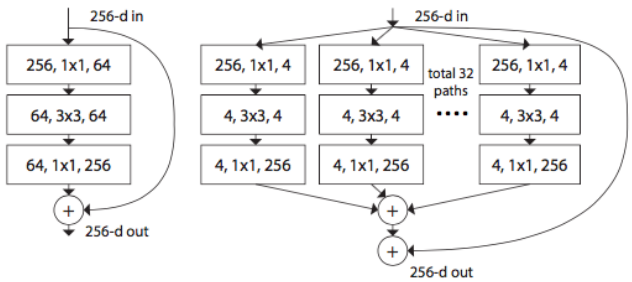


Figure 3. (Left) A ResNet block. (Right) A ResNeXt block with cardinality = 32.

### 3.3. Baseline to Improved: Data Augmentation

To avoid overfitting to train' on Improved, we incorporate standard data augmentation techniques such as; random rotation, brightness, contrast, lighting, and saturation.

## 4. Results

By using train' and validation, now the model (Improved) reflects a better estimate of out-of sample performance, which is shown in Table 4. Improved also shows higher model capacity and less signs of overfitting due to the learning rate scheduling and data augmentation.

| Dataset | @20k | @25k | @30k | @35k |
|---|---|---|---|---|
| train' | 56.83 | 56.90 | 56.65 | 57.13 |
| validation | 44.20 | 51.59 | 51.41 | 51.75 |

Table 4. Improved performance on train' versus validation at various training iterations. All @iterations with mAP

Training curves on Improved for train' loss, validation loss, learning rate, and validation mAP are shown in Figure 4.

Finally, a comparison between Baseline and Improved based on mAP is given in Table 5. For completeness, we show results for Improved with and without data augmentation.

A plot of various evaluation metrics and samples of output from Improved with data augmentation are included in figure 6

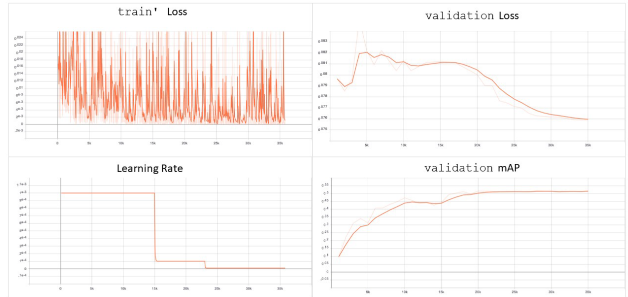


Figure 4. Training curves on Improved.

| Dataset (Source) | Model | Data Augmentation | mAP |
|---|---|---|---|
| Validation | Baseline | No | 44.87 |
| Validation | Improved | No | 46.67 |
| Validation | Improved | Yes | 51.75 |
| test | Baseline | No | 68.85 |
| test | Improved | No | 67.82 |
| test | Improved | Yes | 71.37 |
| test inshore | Baseline | No | 36.22 |
| test inshore | Improved | No | 36.40 |
| test inshore | Improved | Yes | 42.94 |

Table 5. Comparison between Baseline and Improved results.

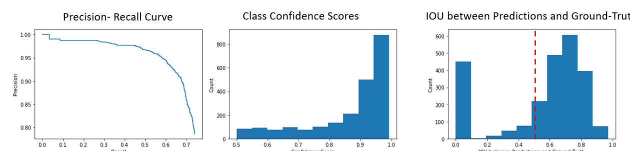


Figure 5. Precision-Recall Curve, Class Confidence Scores, and IOU between Predictions and Ground-Truth obtained from Inference on test.

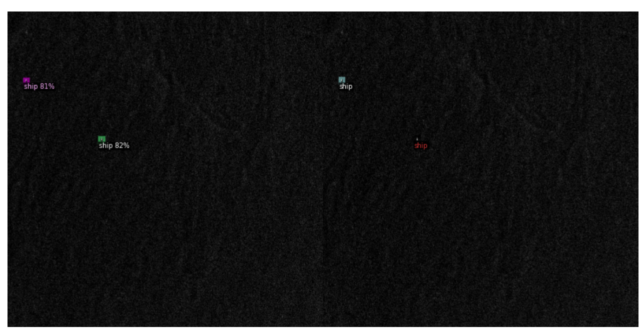

Figure 6. Comparison of Predictions (Left) and Ground Truth (Right) on test offshore

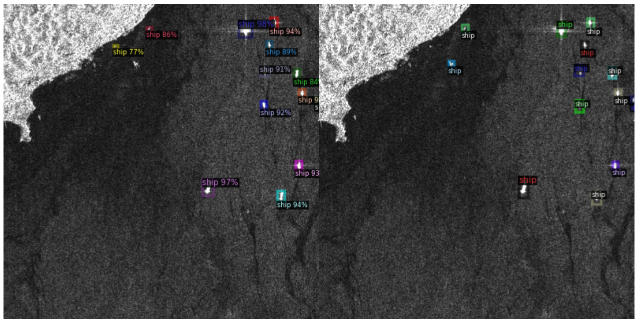


Figure 7. Comparison of Predictions (Left) and Ground Truth (Right) on test inshore with backscattering.

## 5. Discussion and Conclusion

Even though the test inshore performance has improved with Improved, there is still a significant performance difference between the test inshore and test offshore. Due to interference from the surrounding land and backscattering, ships in inshore scenes are harder to detect than those in offshore scenes, which accounts for this observation [1].

It may be possible to enhance the performance of ship detection in inshore scenes by using a Otsu's method, a histogram-thresholding algorithm, to create a Paste-Mask which prevents unrealistic paste locations from occurring. Otsu's method excels for images that have areas of high contrast differences (containing land and sea) like those in test inshore.

Making sure that anchor boxes overlap with such tiny ground truth bounding boxes is the other difficulty we are having when training small object detection models [5, 6]. This can be avoided by using a copy-paste method similar to [6], which augments train' with arbitrary copies of ship annotations. It seems that combining standard + copy-paste data augmentation, as in (+)Improved, is a successful strategy for enhancing test inshore performance.

Despite these advancements, future research on copy-paste with data augmentation could also benefit from more precise sea-land masking. Finally, the results on Improved provide additional evidence that the distributions of the train and test data are distinct. Therefore, it's crucial to make sure that test and validation come from the same distribution and that they both accurately reflect.

## References


[1] Authors. Sar images - principles of remote sensing., 2013. . `https://crisp.nus.edu.sg/ research/tutorial/ sar`$_i nt.html$.

[2] S. Ren K. He, X. Zhang and J. Sun. "deep residual learning for image recognition,", 2020.

[3] Y.Wu K. Tong and F. Zhou. Recent advances in small object detection based on deep learning: A review,, 2020. Image and Vision Computing, vol. 97, p. 103910.

[4] F. Liu S. Yang L. Li Z. Feng L. Jiao, F. Zhang and R. Qu. A survey of deep learning-based object detection,, 2019. CoRR, vol. abs/1907.09408.

[5] J. Murawski J. Naruniec M. Kisantal, Z. Wojna and K. Cho. Augmentation for small object detection, corr, vol. abs/1902.07296, 2019.

[6] N. Mo and L. Yan. Improved faster rcnn based on feature amplification and oversampling data augmentation for oriented vehicle detection in aerial images," remote sensing, vol. 12, no. 16,, 2020.

[7] G. Synnaeve N. Usunier A. Kirillov N. Carion, F. Massa and S. Zagoruyko. End-to-end object detection with transformers,, 2020.

[8] R. Girshick K. He T.-Y. Lin, P. Goyal and P. Dollár. "focal loss for dense object detection,", 2020.

[9] J. Shi T. Zhang, X. Zhang and S.Wei. Hyperli-net: A hyperlight deep learning network for high-accurate and high-speed ship detection from synthetic aperture radar imagery,", 2020. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 167, pp. 123 – 153,.

[10] X. Ke X. Zhan J. Shi S. Wei D. Pan J. Li H. Su Y. Zhou T. Zhang, X. Zhang and D. Kumar. Remote sensing, 2020. "Ls-ssdd-v1.0: A deep learning dataset dedicated to small ship detection from large-scale sentinel-1 sar images.

[11] F. Massa W.-Y. Lo Y. Wu, A. Kirillov and R. Girshick. "detectron2." https://github.com/facebookresearch/detectron2,.

[12] T. Zhang and X. Zhang. High-speed ship detection in sar images based on a grid convolutional neural network,, 2019. Remote Sensing, vol. 11(10).

[13] T. Zhang and X. Zhang. Shipdenet-20: An only 20 convolution layers and ¡1-mb lightweight sar ship detector, 2020. IEEE Geoscience and Remote Sensing Letters, pp. 1–5,.