IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 10 August 2023, accepted 28 August 2023, date of publication 31 August 2023, date of current version 7 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3310539

RESEARCH ARTICLE

A Deep-Learning-Based Lightweight Model for Ship Localizations in SAR Images

SHOVAKAR BHATTACHARJEE[1,2], PALANISAMY SHANMUGAM[1], AND SUKHENDU DAS[2], (Senior Member, IEEE)

[1]Department of Ocean Engineering, Indian Institute of Technology Madras, Chennai 600036, India
[2]Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai 600036, India

Corresponding author: Palanisamy Shanmugam (pshanmugam@iitm.ac.in)

This work was supported by the Prime Minister Research Fellowship (PMRF) Scheme, Government of India, through the PMRF Research Grant.

ABSTRACT Ship detection and localizing its position are indispensable in maritime surveillance and monitoring. Until early 2000, ship detection relied on human intelligence, but with the fast-processing speed, artificial intelligence (AI), especially deep learning, has replaced manual intervention with automatic localization in tracking naval activities. Taking advantage of the continuous and cloud-free ocean observations of Synthetic Aperture Radar (SAR), recent studies have demonstrated some success in utilizing SAR data to localize ships using deep-learning and other AI methods despite the accuracy of the models being lower than the acceptable limit. However, the existing models are inherently complex and time consuming in addition to demanding an extensive computational resource, which pose a significant challenge when applied to satellite-based data. This study presents a computationally efficient deep-learning-based algorithm that has a wider applicability and improves the accuracy over the existing models for ship localization in SAR images. Training and testing of this algorithm were conducted using the SAR Ship Dataset, which contains ship chips with complex backgrounds extracted from Gaofen-3 and Sentinel-1 satellite data. It produced the localized ship's position with bounding boxes in SAR images using the combined traditional computer vision and deep neural network configuration, which comprises a novel backbone network called Ship-Net or S-Net. The S-Net model has a thirteen-layer backbone feature extraction network and a four-layer regression network concatenated. Further, this study proposes a modified combined loss function for optimizing the model performance. A comparative analysis of the proposed S-Net model was done using the various pre-build model architectures and loss function combinations. The results showed that the S-Net model with a combined loss function yielded 94.88% precision and 79.68% recall, with 12.58% precision and 7.39% recall higher than the state-of-the-art Faster RCNN baseline model. The proposed S-Net model has a relatively higher performance than the existing state-of-the-art models for ship localization in SAR images and can become an efficient tool for ship localization in optical images with suitable architectural and training scheme modifications for better coastal surveillance and worldwide naval security.

INDEX TERMS Deep-learning, maritime surveillance, object detection, ship detection, SAR.

I. INTRODUCTION

Ship detection and localization on Synthetic Aperture Radar (SAR) is vital in maritime applications, including fishery monitoring and management, rescue operations, marine transit and traffic surveillance, and national and international security. The International Maritime Organization (IMO) requires all ships to carry Automatic Identification Systems (AIS) for security reasons and to avoid collisions with other ships. However, a practice known as "going dark" (deactivating the AIS system) is a common phenomenon done intentionally near the Exclusive Economic Zone (EEZ).

The associate editor coordinating the review of this manuscript and approving it for publication was Zhongyi Guo.

These territorial and international waters affect levels of maritime security. Due to the potentially promising capabilities of high-resolution/wide swath remote sensing, and all-day and all-weather imaging, ship detection in SAR images has become one of the most effective means in maritime surveillance and many valuable applications for civil and military fields. In recent years, experimental results and the developed methodologies based on the imaging sensors like RADARSAT-2, TerraSAR-X, and Sentinel-1 have demonstrated and promoted the utility of SAR images for ship detection and localization applications [1]. Manual ship localization in satellite images is very laborious and time-consuming due to the comprehensive coverage of the ocean and the requirement of expertise in subjective analysis for labeling the coordinates of ships. The development of statistical algorithm-based localization techniques reduced human effort and improved efficiency. The most popular methods are based on the constant false alarm rate (CFAR) [2], but limitations include suppressing clutter and controlling near-shore false alarms. In recent years, the modified CFAR-based approaches have been proposed [3] along with a variety of different methods, including the generalized likelihood ratio test (GLRT) [4], visual saliency [5], super-pixel segmentation [6], polarization decomposition [7] and some auxiliary features (oil spill clues and ships wake) [8], [9] based localization. Another popular method, template-based detection [10], provides specific target templates according to the different scenarios. However, establishing a template library requires expertise and weak generalization ability [11]. The development of machine learning techniques like AdaBoost [12], decision tree [13], and support vector machine (SVM) [14] increased the efficacy with a considerably reduced processing time, but their accuracy is limited. The recent deep learning techniques have obsoleted the machine learning techniques by solving the localizing problems with high accuracy and faster performance.

Since object detectors in deep learning can learn the target positions directly from the raw dataset with annotation, it has been adopted for ship localization in SAR images. Transfer learning and fine-tuning are extensively used in this process. The traditional process of ship localization using an object detector is to modify a pre-trained model like RCNN [15], Fast RCNN [16], Faster RCNN [17], SSD [18], and YOLO [19] and train them with SAR image chips. These pre-trained state-of-the-art deep learning models cannot maneuver the SAR characteristics because the pre-trained object detectors are trained on datasets like ImageNet [20] or MS COCO [21], which have very different feature characteristics than SAR images. SAR is a coherent imaging process with characteristics like foreshortening, layover, and shadowing, which are not found in RGB images. The existing models are very complex in nature and require a large computational resource. Hence, there is a huge demand for a lightweight model that overcomes the above issues.

Recently, a few lightweight models have been proposed by various researchers to reduce the computational cost. For instance, Xe et al. [22] proposed a Lite-YOLOv5 that reduces the computational cost of the original YOLOv5 by 56.59%. Wang et al. [23] proposed a detection module from the fisher vectors that suppress sea clutter and enhances ship detection by introducing two new global and one improved local cue. Geng et al. [24] proposed a lightweight CNN based module that separates false alarms near the target object. Miao et al. [25] proposed an improved lightweight RatinaNet model by replacing the backbone with a ghost module, and Xiong et al. [26] proposed a lightweight model by redesigning and optimizing pyramid pooling in the YOLOv5n model. Most of these models still are complex and computationally expensive. Here, a new state-of-the-art algorithm has been proposed with end-to-end training to localize ships in SAR images. This algorithm is less complex and reduces the computational cost and resources.

This study aims to design a network pipeline to reduce computational complexity and increase ship localization accuracy with SAR images. The proposed algorithm is a single-stage object detector with a novel backbone feature extraction network and a regression network concatenated for localizing the ship chips in the SAR images. A modified combined loss function has also been proposed to optimize the learning process and improve accuracy. This study will be very beneficial in coastal surveillance for automatically monitoring accurate ship movements in the ocean without human intervention. Accordingly, the main contribution of this work is organized as follows: i) a comprehensive review of popular ship localization techniques based on satellite remote sensing data, ii) a detailed description of the proposed state-of-the-art model and loss function developed for ship localization, iii) a comparison of the experimental results for the proposed state-of-the-art model with existing models, and iv) a brief discussion on further challenges, failure cases, and possible futuristic trends.

II. DATASET

The SAR dataset used in this study are those reported by Wang et al. [27]. The dataset contains 39,716 ship chips (an updated version reported by them in 2021 after correcting the bounding box errors and removing the repeated clips) from 102 Gaofen-3 satellite images and 108 Sentinel-1 satellite images. For Gaofen-3 data, the image resolutions are 3 m, 5 m, 8 m, and 10 m with Ultrafine Strip-Map (UFS), Fine Strip-Map 1 (FSI), Full Polarization 1 (QPSI), Full Polarization 2 (QPSII) and Fine Strip-Map 2 (FSII) imaging modes, respectively. For Sentinel-1 data, S3 Strip-Map (SM), S6 SM, and IW-mode were used. The ship chips are 256 × 256 pixels, with one or more ships in each chip with a multiscale resolution. The annotation was done by SAR experts using a LabelImg software. In the field of ship detection and recognition in SAR imagery, several popular datasets have been used by the researchers, including SSDD [28], HRSID [29], SRSDD-v1.0 [30] and LL-SSDD-v1.0 [31]. For this study, SAR Ship Dataset was chosen due to its distinct advantages. The SAR Ship Dataset offers a consistent resolution across all images, providing a standardized testing environment. Additionally, it contains a large number of images, allowing for a comprehensive representation of real-world maritime scenarios. The latest version released in 2021 has been used for the proposed model with stronger augmentations.

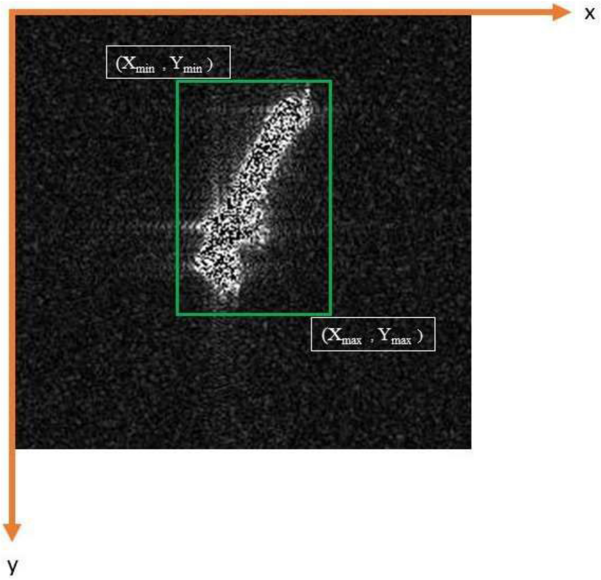

FIGURE 1. Sample image extracted from the SAR ship dataset.

The dataset uses the PASCAL VOC format to explain the annotations. The initial release (2019) adopted this format. As shown in Fig. 1, each ship chip in an image has a bounding box (BBox) around it, denoted by the letters X and Y coordinates. Each BBox has a ship center denoted by X_0 and Y_0, and the ship height and width (H & W) indicate the ship's location.

$$H = Y_{max} - Y_{min} \tag{1}$$
$$W = X_{max} - X_{min} \tag{2}$$
$$X_0 = (X_{max} + X_{min})/2 \tag{3}$$
$$Y_0 = (Y_{max} + Y_{min})/2 \tag{4}$$

here, X_{max} and X_{min} are the maximum and minimum range of ship pixels in the X direction, and Y_{max} and Y_{min} are the maximum and minimum range of ship pixels in the Y direction. The number of channels in RGB images is three by default, which are denoted by the letters R, G, and B (Red, Green, and Blue). Since SAR images only have a gray-level channel, it is copied twice to obtain RGB-3 channel images. The entire dataset was randomly split into training (70%), validation (20%), and test (10%) datasets. After a careful analysis of the data, it was found that the dataset contains mainly three types of ship chips (a) Small-sized ships in the unenhanced and enhanced form in the open ocean, (b) Large-sized ships in the unenhanced (non-contrast) and enhanced forms in the open ocean, and (c) Ships without land cover in the coastal area. This gives the benefit of detection for multiscale ship images.

III. METHODOLOGY

This section presents the theoretical basis of the proposed algorithm for ship localization. Generally, the ship localization algorithms are based on the traditional computer vision approaches and divided into two categories: (a) two-stage detectors and (b) one-stage detectors. The two-stage detectors are efficient, but it takes a long processing time. It has two neural networks, one creating region proposals, and the following selects the best suitable region proposal, localizing the ships. One stage detector does not generate region proposals but regresses bounding box coordinates for the target ships for localization in the image. As mentioned earlier, the single-stage algorithm adopted for this study as one-stage detectors improves the speed and reduces the computational redundancy compared to the two-stage detectors algorithms.

The single-stage algorithm and the detailed network architecture of the proposed model are shown in Fig. 2 and Fig. 3. This model has two concatenated networks (Fig. 3) and works as a single model with the feature extraction and regression networks (working together as S-Net). The feature extraction network extracts high-level features from the image, and the regression network predicts the bounding box coordinates for the target position from the extracted features. The loss function calculates the difference between the predicted and annotated coordinates (Fig. 2) and feedbacks the information to the networks to further update the weights. The architecture and loss function are discussed in the following section.

A. MODEL ARCHITECTURE AND TRAINING

This section describes the architectural design of the proposed model and training scheme used with optimization in this study.

1) FEATURE EXTRACTION NETWORK

The feature extraction network (as shown in Fig. 3) is a deep learning-based convolutional neural network that has eleven layers of feature extraction network with seven convolutional layers with a filter size of 3 × 3 and with channel numbers of 32, 64, 128, 256 and 512 and four max-pooling layers with a filter size of 2 × 2 with stride = 1. It has an input layer of 256 × 256×3 on top of the first convolutional layer and a flatten layer after the last max pooling layer. Thirteen layers in total have been used to design the feature extraction network. The convolutional layer applies a convolutional operation with the pixel brightness values, and the max pooling layer pools the maximum value among the filter size. A flattened layer creates a new array with all sub-array elements concatenated to a single row as a high-level feature. The Sigmoid and ReLU activation functions have been used for this network. A skip connection has been established from third to flatten

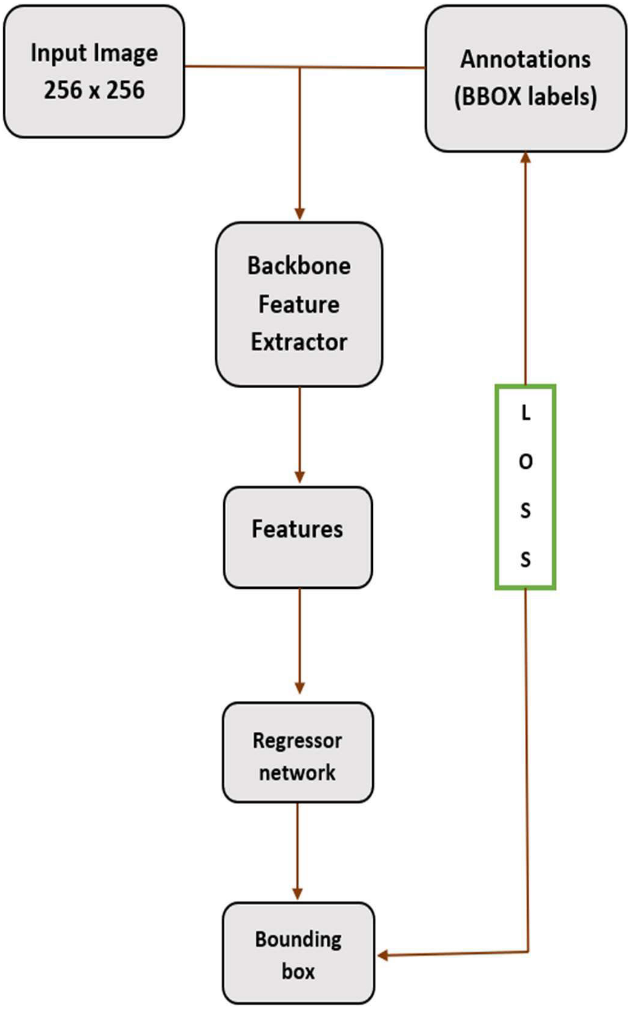

FIGURE 2. Schematic diagram of the proposed algorithm for ship localization.

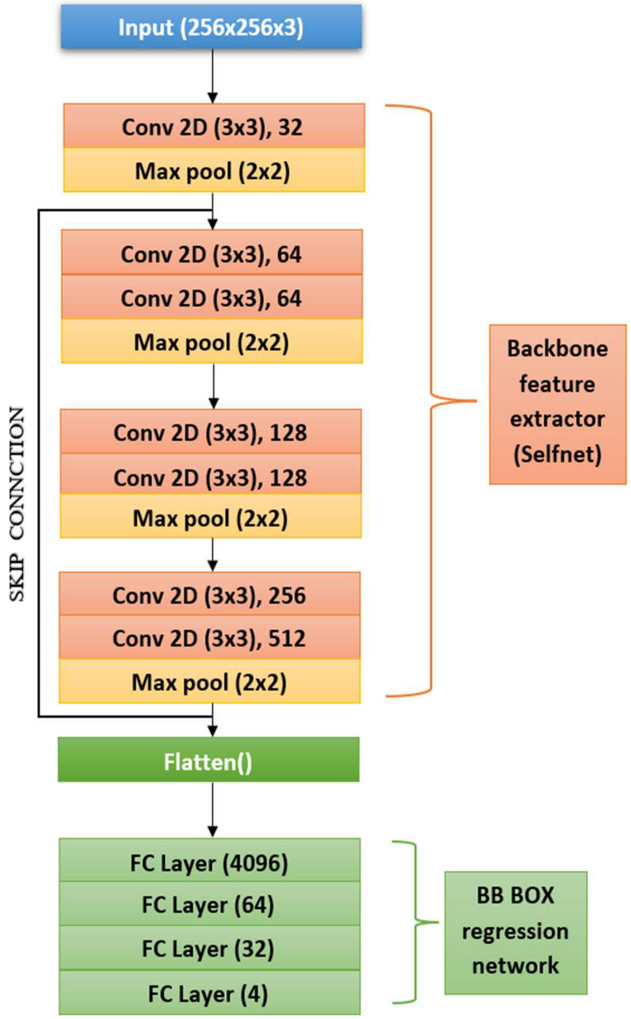

FIGURE 3. The detailed architecture of the proposed S-Net model.

layer to focus on large features along with deep features for better accuracy.

2) REGRESSION NETWORK

The regression network learns a relationship between the features and the target. The network input includes the high-level features from the flattened layer, and it predicts bounding box coordinates through the relationship between the features and the ground truth. The network has four fully connected convolution layers with 128, 64, 32 & 4 neurons, respectively, and the ReLU and sigmoid activation functions have been used for threshold firing. The final layer predicts the bounding box coordinates of the target ship position.

3) LOSS FUNCTION

The loss function is a mathematical function that computes the difference between the model output and the ground truth.

The model calculates and updates its weights based on errors. Different loss functions are used for general object detection, but the two most popular loss functions are used in these experiments, along with a modified loss function, include Mean Square Error loss and Huber loss functions. The Mean Square Error (MSE) loss function takes the form

$$\mathbf{L}_{\mathbf{mse}} = \frac{1}{\mathrm{N}} \sum\nolimits_{i=1}^{\mathrm{N}} \left[\left(y_i - \tilde{\mathrm{y}}_i\right)\right]^2 \quad (5)$$

where y_i = actual value, $\tilde{\mathrm{y}}_i$ = predicted value, and N is the number of images. The Modified Mean Square Error is a Mean Square Error loss function with an added optimization parameter (λ) to optimize the model performance.

$$\mathbf{L'}_{\mathbf{mse}} = \frac{1}{\mathrm{N}} \sum\nolimits_{i=1}^{\mathrm{N}} \left[\lambda\left(\mathrm{y}_i - \tilde{\mathrm{y}}_i\right)\right]^2 \quad (6)$$

where λ = loss coefficient (empirical value), and $0.01 < \lambda < 1$, with 0.05 interval for model testing, $\mathrm{y_i}$ = actual value, and $\tilde{\mathrm{y}}_\mathrm{i}$ = predicted value. The Huber loss function is given by

$$\begin{aligned}\mathbf{L}_{\delta} &= \left\{\frac{1}{2}\left(\mathrm{y}-\mathrm{f}\left(\mathrm{x}\right)\right)^{2}\right\} \quad \text{for } |\mathrm{y}-\mathrm{f}\left(\mathrm{x}\right)| < 0 \\ &= \delta\left|\mathrm{y}-\mathrm{f}\left(\mathrm{x}\right)\right| - \frac{1}{2}\delta^{2} \text{ otherwise} \end{aligned} \tag{7}$$

where y = predicted value, f(x) = actual value, and δ is the hyperparameter whose value is taken as 1 for this model. It limits the influence of poorly fitting data points in the model.

The final loss function is a combined loss function is based on the Modified Mean Square Error and Huber loss functions and works as a regression loss

$$\mathbf{L}_{\mathbf{final}} = \mathbf{L'}_{\mathbf{mse}} + \mathbf{L}_{\delta} \tag{8}$$

The combined loss function produces better results for all the experiments conducted with S-Net, which makes the model more efficient.

B. PERFORMANCE ASSESSMENT

The model's efficiency was calculated based on the Intersection Over Union (IOU) value – a ratio between the overlap and union of two bounding boxes (i.e., predicted and ground truth boxes (as shown in Fig. 4) and five statistical metrics (1) Precision, (2) Recall, (3) F1 score, (4) Accuracy, and (5) Accuracy boost. These matrices were obtained on metric threshold calculations based on the IOU values. The thresholds are True Positive (TP), True Negative (TN), and False Positive (FP), as shown in table 1. If the IOU is equal to or greater than 0.5, it is called a True Positive, and it means good localization. If the IOU is between 0 and 0.5, it is called a True negative and means partial localization that cannot be considered a ship. The output target is shifted from the actual ship. If the IOU is 0, it is called a False Positive, which means the model predicts a target that is entirely not a ship.

Based on the threshold, the matrices are evaluated as:

1) Precision: It is a measure of correct prediction. A fraction of predictions conveys the percentage of correct predictions among true predictions. The precision metric is calculated as

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{TN}}$$

2) Recall: It is a measure of correct predictions among all positive predictions. It is calculated as

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

3) F1 score: F1 score is a harmonic mean of both precision and recall. It is used to compare and evaluate the model performance. The F1 score is calculated as

$$\text{F1 Score} = 2\text{x}\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4) Accuracy: It refers to correct predictions among all the prophecies. It's a measure of the model's efficiency and can be calculated as

$$\text{Accuracy} = \frac{\text{TP}}{\text{TP} + \text{TN} + \text{FP}}$$

5) Accuracy boost: It is a relative measurement of the accuracy of the models against the baseline model. It is used to compare the effectiveness of models with a baseline model. It can be calculated as:

$$\text{Accuracy boost} = \frac{\text{Accuracy}}{\text{Accuracy (Baseline model)}}$$

This metric is used to compare the performance of the proposed model with the baseline data. The proposed model boosts its performance by 124% compared to the baseline model (table 2).

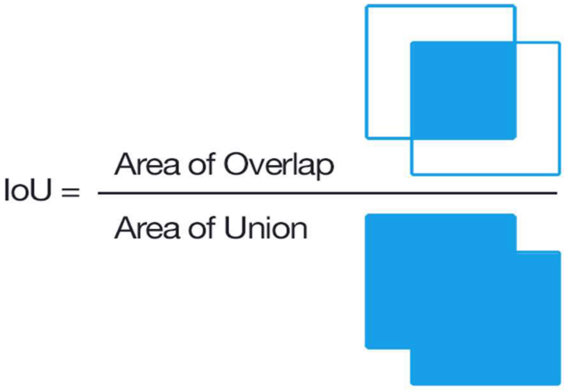

FIGURE 4. Intersection over union (base for accuracy metrics).

TABLE 1. Metric threshold calculations based on the IOU data.

Metric	IOU
TP	IOU >= 0.5
TN	0<IOU<0.5
FP	IOU = 0

IV. EXPERIMENTAL DETAILS

Several experiments were conducted with the combination of pre-trained models that contain different backbone architectures and different loss functions, such as Faster RCNN with its own loss function, SSD with its own loss function, VGG16 with the MSE, Huber and combined loss functions for both ImageNet pre-trained and end-to-end trained weights, Resnet50 with the MSE, Huber and combined loss functions with ImageNet and end-to-end trained weights, ResNet152 with the MSE, Huber combined loss functions with end-to-end trained weights, Inception with the MSE loss function, and proposed S-Net with the MSE, Huber and combined loss functions (modified MSE with Huber) with end-to-end training weights. The Faster RCNN was considered a baseline model due to its popularity as an object detection benchmark algorithm in computer vision. The existing models, such

TABLE 2. Experimental results of the proposed and baseline model.

Model	Loss Function	Precision	Recall	F1 Score	Accuracy	Accuracy Boost
Faster RCNN (Baseline)	Faster RCNN	82.3	72.29	76.97	0.62	1
S-Net	MSE + Huber	94.88	79.68	86.61	0.77	1.24

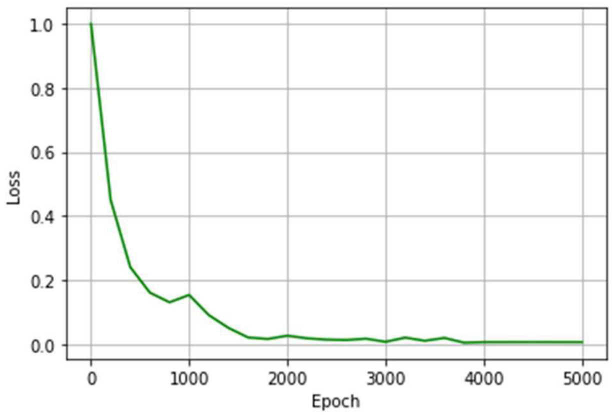

FIGURE 5. Training loss vs epoch graph for the convergence of S-Net model.

as VGG16, ResNet50, ResNet152, and Inception V3, were modified in their last layer, replacing the last layer with four fully connecting layers for localization operation. The present study used the Google Colab platform with Python 3.8 support under the Colab environment to train the models. The Nvidia Tesla P100 GPU with tensor- flow 2.8 as the backend and CUDA 11.0 and cuDNN 7.0.5 were used for training purposes. The programing language CUDA was used for GPU training, and the parallel computing architecture cuDNN was used for faster training. The max iteration epoch was 4000, which was found suitable for the convergence of the propose model (Fig. 5) with a batch size of 32 and a learning rate of 0.001 with the Adam optimizer with Train, Validation and Test ratio as 7:2:1. The training automatically stops if the training loss converges below 0.001.

V. RESULTS AND DISCUSSION

This section presents the performance evaluation results based on several statistical matrices calculated between the model predictions and ground truth data and a rigorous comparative analysis to verify the model's capability. The model results are evaluated quantitatively and qualitatively using the dataset. The proposed algorithm is based on convolutional neural network (CNN) architectures where the convolutional network actively extracts more complex features with increasing network depth due to the added nonlinearities in activation functions like ReLU and Sigmoid. To evaluate the proposed model, seven CNN models were chosen with different loss functions (referred in section V) because of their wide performance variability and a state-of-the-art Faster RCNN model was considered the benchmark baseline model. The results of the proposed models on ship localization in SAR images are shown in table 2. It is observed that the proposed S-Net model exceeds the baseline Faster RCNN model in all the accuracy matrices. The values of Precision, Recall, and F1-Score especially prove that the S-net model can improve the ship localization performance with different scales simultaneously. The Precision, Recall, and F1 Score metrics are 12.58, 7.39, and 9.64 points which are higher than those of the baseline Faster RCNN model. It indicates that the S-Net model can predict more precise ship locations than the baseline model. The S-Net boosts the accuracy by 1.24 times higher than the baseline model, which indicates a significantly increased overall performance compared to the Faster RCNN baseline model. Table 4 compares the performance of the proposed S-Net model with existing models on the dataset. The proposed S-Net model with the combined loss function has the best results of 94.88 precision points and shows a better capability than other existing models. The benchmark state-of-the-art models like Faster RCNN and SSD have poorer performance than the latest models like ResNet50, VGG 16, ResNet152, and Inception, because of a significant mislocalization of ships on SAR images. In this comparison, the S-Net, VGG16, and ResNet152 models yielded substantially more than 90 points in precision, indicating a high-performance ability for ship localizations in SAR images. However, the performance of these models varies depending on the loss function, as shown in table 4. In general, models with the combined loss function performed better than other loss functions. The S-Net model with the combined loss function produced the best results on the dataset. Our results demonstrate that this model can efficiently handle the localization of ships in SAR images, and its performance surpasses other best-performing models.

The second-best model, ResNet152, with the combined loss function, performed slightly low by 0.96 precision points. Yet its computational complexity is very high due to a very deep network architecture, where VGG16 with combined loss function has a close computational cost but

TABLE 3. Comparison of the models' computation speed based on the test image.

Sl. No.	Model	Layer Depth	FPS	No. of Parameters
1	Faster RCNN	50	6	33,193,587
2	SSD	16	16	24,837,492
3	Inception	51	43	31,250,564
4	ResNet50	53	58	40,375,524
5	ResNet152	155	37	75,158,756
6	VGG 16	19	76	18,919,588
7	YOLOv5n	16	70	1,934,675
8	**S-Net**	**15**	**83**	**1,318,468**

shows a less performance of 2.08 precision points. The S-Net has the lowest computational cost comprising the lowest network depth and parameter (table. 3) and producing the best performance, which makes the model more affordable for ship localization in SAR images. Faster RCNN produced a precision score of 82.3 points and an overall accuracy of 0.62 with many false positive cases. Our analysis shows that most false positive cases occurred with ships near the harbor or coastal area (or land cover near the ships). It indicates that the model poorly differentiated ships from the land features in the coastal regions. The SSD model yielded an overall accuracy of 0.53 and a precision of 72.4 points, with 9.9 points lower than the Faster RCNN model. The false positive cases produced by this model increased in comparison to those produced by the Faster RCNN model, which indicates the least performing model (table 4). The modified Inception model produced precision points 73.6 and 75.68 with the MSE and combined loss function, which are 8.7 and 6.62 points lower than the baseline model and 1.2 and 3.28 points higher than the SSD model. Moreover, it is computationally complex due to a very deep network architecture and yields increasingly lower accuracy with the increasing network depth. The qualitative results showed that most of the true negative and false positive cases have occurred in noisy images (Fig. 6). Thus, the model was unable to differentiate between ships and noise features. In contrast, the ResNet152 model (end-to-end trained) with MSE, Huber and combined loss functions produced precision scores of 92.92 and 93.88 and 93.92, which are 10.62 points and 11.58 and 11.62 points higher than those of the baseline model. The ResNet50 model produced 84.95, 85.9, 89.42, and 90.25 precision scores, respectively, when implemented with i) ImageNet weights and MSE loss function ii) End-to End trained weights and MSE loss function, iii) End-to-End trained weights and Huber loss function, and iv) End-End weight with combined loss function with the precision score of 2.65, 3.6, 7.12, and 7.95 points higher than those of the baseline model and 8.93, 7.98, and 4.46 points lower than the ResNet152 model with Huber loss function and 3.67 difference has been observed between these two models with the combined loss function. The VGG16 model produced 80.2, 93.36, and 92.5, and 92.8 precision scores respectively when implemented with i) ImageNet weights and MSE loss function, ii) End-to-End trained weights and MSE loss function, iii) End-to-End trained weights and Huber loss function, iv) End-to-End trained weights and combined loss function with the precision scores of 2.1 points lower and 11.06, 10.2 and 10.5 points higher than the baseline model. The YOLOv5n model yields 87.80 and 79.00 precision and recall points that are 5.5 and 6.71 higher than baseline model and 7.08 and 0.68 lower than those of the proposed model with the combined loss function. The proposed S-Net model with only thirteen layers in its feature extraction network and the lowest computational complexity compared to other models produced the precision scores of 92.01, 90.56, 94.04, and 94.88 points respectively when implemented with the MSE loss function, Huber loss function, Modified MSE loss function $\lambda = 0.1$) and combined loss function. It achieved 9.71, 8.26, and 12.58 precision points higher than the baseline model. In general, it is observed that the combined loss function yields better results than both MSE and Huber loss functions. The model performance was further assessed with the FPS metric, which can be estimated as

$$\text{FPS} = \frac{1}{T}$$

where T is the localization time for the test images. Based on the results from table 3, it can be seen that S-Net can localize ships in test images $\sim$ 14 times faster than the baseline model and nearly three times more quickly than the acceptable limit of 30 FPS, which is followed by the VGG16 model. The computational speed depends on a model's computational cost which is dependent of model's network depth, that gives the proposed model a vast advantage in processing the SAR images faster than the existing models as it has lowest network depth (shown in table 3). Fig. 6 shows the typical ship localization results from some representative images for various scenarios: (A) ships in noisy images, (B) small-sized ships in clear images, (C) large ships, (D) ships in coastal areas, and (E) ships with non-ship objects (model outputs shown from column A to E).

For each sample, every row is the respective model output, and most below is the ground truth. The model-predicted location of the ship is marked by a blue bounding box and its ground truth location by a green bounding box (each box representing a single ship). In most cases, ships in clear images (open ocean scenarios) are detected by the models in close agreement with ground truth data, except the faster RCNN, SSD, and Inception models produce false alarms due to poor discrimination of ships from the non-ship objects. The model performance could be easily affected by certain linear or non-linear or polygon or other object forms or features other than the ships, which would make the models inefficient in the presence of these features. Large ships are easily localized by all models due to their distinct spatial feature information. The images in column A represent the models' critical capability to distinguish between the noise

TABLE 4. Quantitative comparison of the models based on the accuracy metric.

Sl. No.	Model	Loss Function	Precision	Recall	F1 Score	Accuracy	Accuracy Boost
1	**Faster RCNN (Baseline)**	**Faster RCNN**	**82.3**	**72.29**	**76.97**	**0.62**	**1**
2	SSD	SSD	72.4	66.66	69.41	0.53	0.85
3	VGG 16 (ImageNet)	MSE	80.2	69.29	74.34	0.60	0.96
4	VGG 16 (End to End trained)	MSE	93.36	77.86	84.90	0.74	1.19
5	VGG 16 (End to End trained)	Huber	92.5	79.28	85.38	0.75	1.20
6	VGG 16 (End to End trained)	MSE+Huber	92.8	79.36	85.55	0.76	1.22
7	Resnet 50 (ImageNet)	MSE	84.95	74.15	79.18	0.66	1.06
8	Resnet 50 (End to End trained)	MSE	85.9	74.78	79.95	0.67	1.08
9	Resnet 50 (End to End trained)	Huber	89.42	74.78	81.44	0.70	1.12
10	Resnet 50 (End to End trained)	MSE+Huber	90.25	75.20	80.04	0.72	1.16
11	Inception	MSE	73.6	67.4	70.36	0.55	0.88
12	Inception	Huber	73.00	68.22	70.52	0.54	0.87
13	Inception	MSE+Huber	75.68	69.12	72.25	0.57	0.91
14	Resnet 152 (End to End trained)	MSE	92.92	78.17	84.90	0.74	1.19
15	Resnet 152 (End to End trained)	Huber	93.88	78.68	85.61	0.76	1.22
16	Resnet 152 (End to End trained)	MSE+Huber	93.92	78.60	85.57	0.76	1.22
17	YOLOv5n	YOLO	87.80	79.00	83.16	0.68	1.09
18	S-Net	MSE	92.01	78.4	84.66	0.74	1.19
19	S-Net	Huber	90.56	77.73	83.65	0.73	1.17
20	S-Net	Modified MSE $(\lambda = 0.1)$	94.04	80.70	86.85	0.76	1.22
21	**S-Net**	**MSE+Huber**	**94.88**	**79.68**	**86.61**	**0.77**	**1.24**

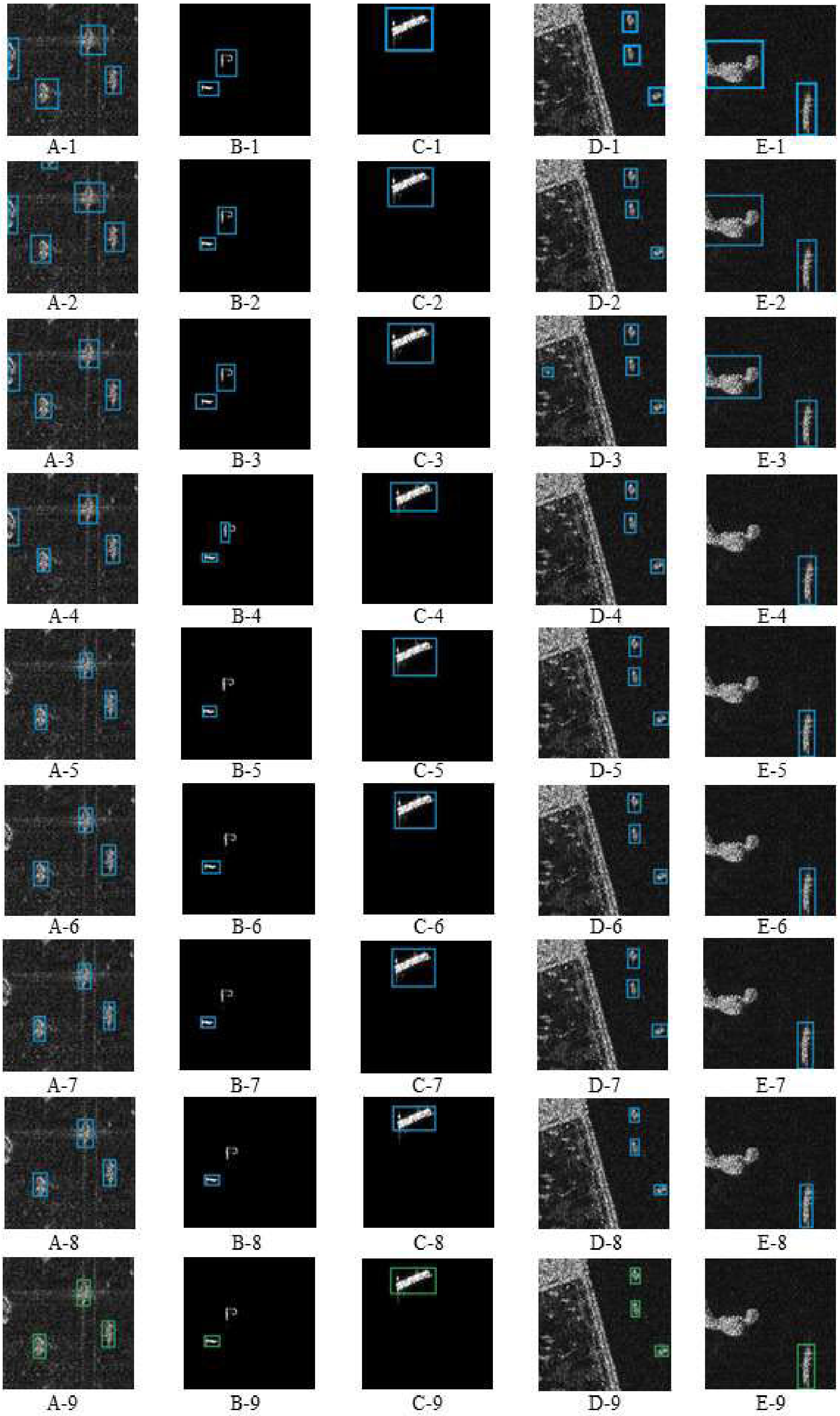

FIGURE 6. **The qualitative results for the implemented models. (A-E)-1 represents the outputs of the Faster-RCNN model; (A-E)-2 represents the outputs of the SSD model; (A-E)-3 represents the outputs of the Inception model with combined loss function; (A-E)-4 represents the outputs of the ResNet50 model with End-to-End weights and combined loss function; (A-E)-5 represents the outputs of VGG16 model with End-to End weights and combined loss function; (A-E)-6 represents the outputs of ResNet152 model with End-to End weights and combined loss function; (A-E)-7 represents the outputs of YOLOv5n model; (A-E)-8; represents the outputs of S-NET model with End-to-End weights and combined loss function and (A-E)-9 represents the Ground truth.**

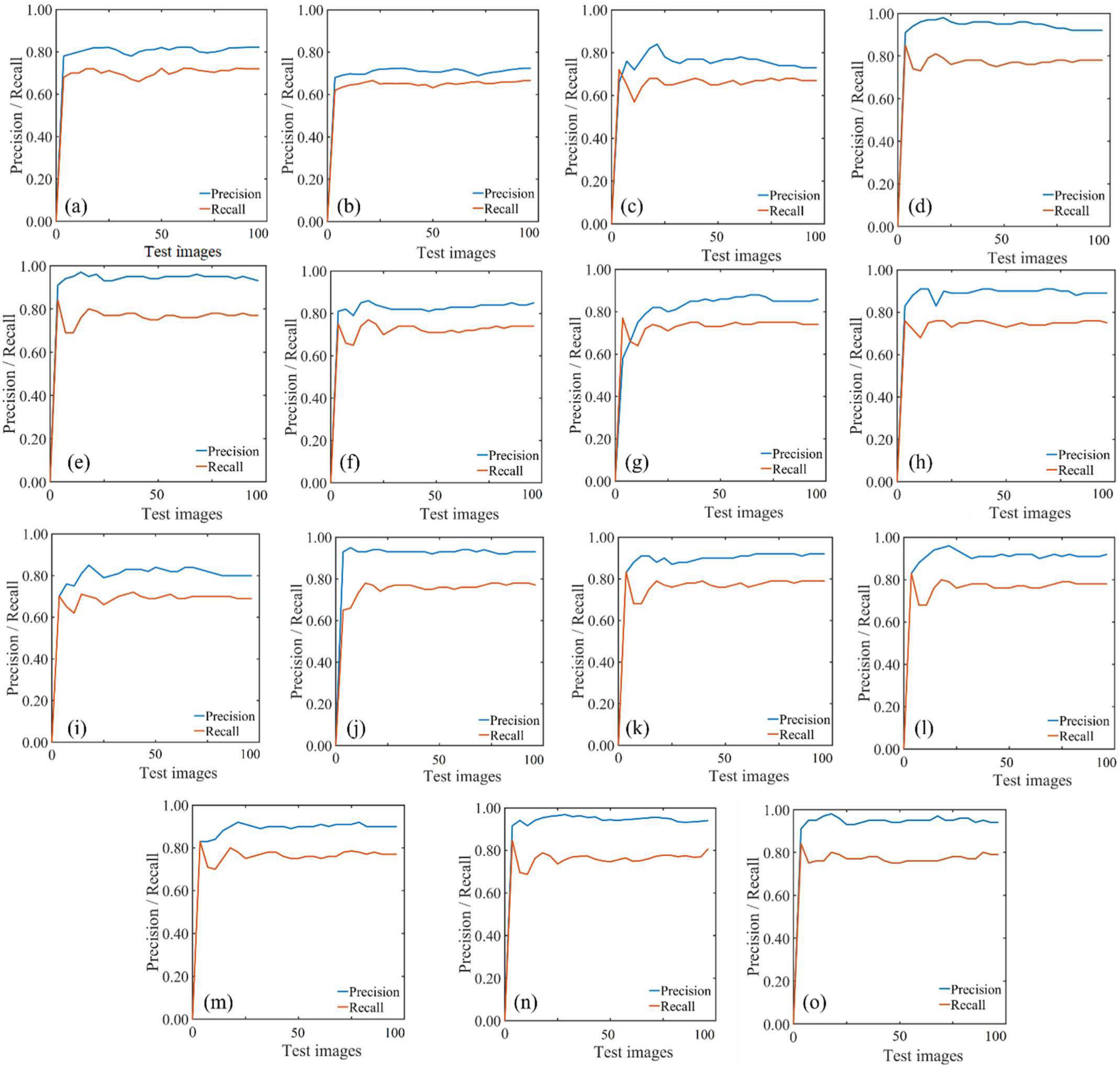

FIGURE 7. Precision/Recall vs. Test images for (a) Faster RCNN; (b) SSD; (c),(d) and (e) VGG16 with MSE (ImageNet), MSE (E-2-E), combined loss function (E-2-E); (f),(g) and (h) ResNet50 with MSE (ImageNet), MSE (E-2-E), and combined loss function(E-2-E); (i) Inception with combined loss function; (j) and (k) ResNet152 with MSE (E-2-E) and combined loss function (E-2-E); and (l) YOLOv5n; (m), (n) and (o) Huber loss function, MMSE, and combined loss function model.

and ship features due to the less indistinguishable spatial information of the foreground and background features in the noisy images. In all these cases, the S-Net model is most efficient in localizing the ships, followed by the ResNet152 model. The results showed a critical failure of the Faster RCNN, SSD, Inception, and ResNet50 models due to the mislocalization of ships in these noise images. Both the VGG16 and S-net models produced nearly identical results closely consistentwith ground truth data. These two models are robust because of their less mislocalization of ships or false alarms, compelling performance in a noisy environment, and prediction of bounding boxes closest to the ground truth data. However, the S-Net model is most efficient for ship localization in SAR images due to its higher performance (in terms of accuracy and processing speed) bounding boxes closest to the ground truth data. However, the S-Net model is most efficient for ship localization in SAR images due to its higher performance (in terms of accuracy and processing speed) and lower computational complexity. Fig. 7 shows the variation of precision and recall metrics of the different models tested on several SAR images. Blue and brown lines indicate the precision and recall curves respectively.

TABLE 5. False positive rate or sensitivity of the model.

Sl. No.	Model	Loss Function	Sensitivity (↓)
1	Faster RCNN (Baseline)	Faster RCNN	0.360
2	SSD	SSD	0.432
3	VGG 16 (ImageNet)	MSE	0.357
4	VGG 16 (End to End trained)	MSE	0.200
5	VGG 16 (End to End trained)	Huber	0.235
6	VGG 16 (End to End trained)	MSE+Huber	0.230
7	Resnet 50 (ImageNet)	MSE	0.337
8	Resnet 50 (End to End trained)	MSE	0.326
9	Resnet 50 (End to End trained)	Huber	0.271
10	Resnet 50 (End to End trained)	MSE+Huber	0.261
11	INCEPTION	MSE	0.426
12	INCEPTION	Huber	0.425
13	INCEPTION	MSE+Huber	0.423
14	Resnet 152 (End to End trained)	MSE	0.214
15	Resnet 152 (End to End trained)	Huber	0.191
16	Resnet 152 (End to End trained)	MSE+Huber	0.198
17	YOLOv5n	YOLO	0.213
18	S-Net	MSE	0.239
19	S-Net	Huber	0.266
20	S-Net	Modified MSE ($\lambda = 0.1$)	0.209
21	**S-Net**	**Modified MSE ($\lambda = 0.1$) + Huber**	**0.174**

The Faster RCNN (Fig. 7(a)) model shows a variation in both precision and recall curves for a number of test images, indicating the abrupt misclassification of false alarms generated by the model. Similar results were obtained for the SSD model (Fig. 7(b)), with a lesser amplitude showing a lower performance than the baseline Faster RCNN model. A spike in precision and fall in recall at the beginning of the curve indicates a decrease in mislocalization and a sudden increase in false alarms by most models except Faster RCNN and SSD. The Inception model shows a wide variation in both precision and recall curves, similar to the VGG16 model (Fig. 7(c)). The ResNet152 model with an MSE loss function has a very stable constant and a higher precision curve. Noticeably, the recall curve for the S-Net model with a combined loss function falls initially (like VGG16 and ResNet50 models) and spikes up subsequently, which reduces the false alarm. The nano YOLOv5 model (YOLOv5n) has a similar characteristic to VGG16 (Fig. 7(l)), containing a wide peak in the beginning and a sharp fall in recall. In contrast, the S-Net with a combined loss function improved the results with the highest amplitude and a more stable recall curve compared to other models (Fig. 7(o)).

A. FAILURE CASES

Figure 8 demonstrates the failure cases of the proposed S-Net model, whereas Figure 8 (a) and (b) represents false alarm predictions of the ships in the coastal area; Figure 8(c) represents mislocalization of the small ships in noisy images, and Figure 8(d), (e), and (f) represents ground truth data. The model failed to discriminate between ship and land pixels in the first two cases. In the third case (Fig. 8(c)), the model failed to distinguish between noise points and ship intensity points on a noisy image, but it distinguished similar types of small ships with no noise points (Fig. 6-B(8)). Though the number of false alarms and misclassification are very low, this failure still leads to the conclusion that the model needs more sample images or strong augmentation in the dataset to solve the failure scenarios. Large ships are easily localized by all models due to their distinct spatial feature information. The images in column A (Fig. 6) represent the models' critical capability to distinguish between the noise and ship feature due to the less indistinguishable spatial information of the foreground and background features in the noise images. In all these cases, the S-Net model is most efficient in localizing the ships, followed by the ResNet152 model. The results showed a critical failure of the Faster RCNN, SSD, Inception, and ResNet50 models due to the mislocalization of ships in these noise images. Both the VGG16 and S-Net models produced nearly identical results closely consistent with ground truth data. These two models are robust because of their less mislocalization of ships or false alarms, compelling performance in a noisy environment, and prediction of bounding boxes closest to the ground truth data. However, the S-Net model is most efficient for ship localization in SAR images due to its higher performance (in terms of accuracy and processing speed) and lower computational complexity.

B. SENSITIVITY ANALYSIS

The sensitivity of a model, also known as the false positive rate of a model, refers to how much a model is sensitive in generating a false alarm. The lower the false alarm rate of the model, the lesser the sensitivity and more stability. The false

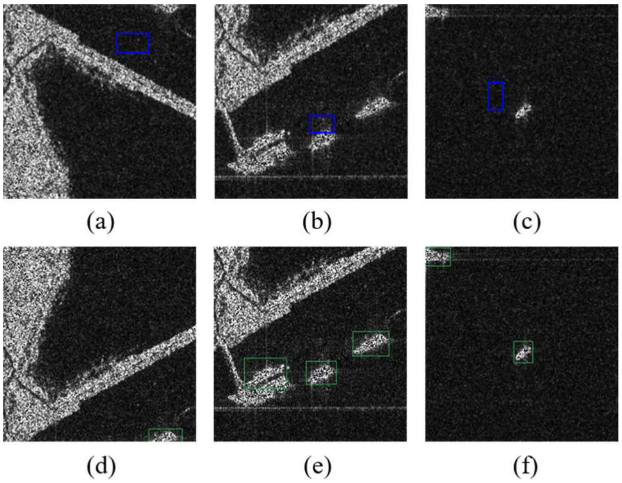

FIGURE 8. Predicted failure case false alarms for ships near the coastal area with land cover (a and b), small ships in noisy images (c) and the corresponding ground truth from the dataset (d-f).

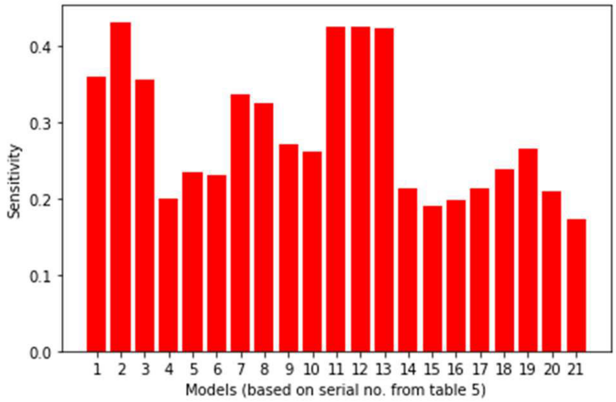

FIGURE 9. Column chart for sensitivity analysis (The x axis labels are models with serial number described in table 5).

positive rate is the ratio of true negatives divided by the sum of true negatives and false positives.

Table 5 shows the sensitivity of various models, where the S-Net model with a combined loss function is the least sensitive, and SSD is most sensitive for this test dataset. The VGG16 model with end-to-end training with MSE and combined loss functions and the ResNet152 model with end-to-end training with Huber and combined loss functions are highly stable compared to the baseline model but more sensitive (2.6%, 5.6% and 1.7%, 2.4%) as compared to the S-Net model with the combined loss function. The lightweight nano YOLO model is 1.3% less sensitive than the VGG16 model with the MSE loss function (E-2-E) and 3.9% more sensitive than the S-Net model with a combined loss function. It is noticeable that the combined loss function increases stability in all the models except ResNet152 and VGG 16 with MSE (E-2-E). Fig. 9 show the bar chat comparing sensitivity of the models according to table 5. These results indicate that the S-Net model with a combined loss function can be considered the robust model for ship localization in SAR images.

VI. CONCLUSION

Ship localization is critical for maritime surveillance and other applications. The present study reported a state-of-the-art architecture and a deep learning-based algorithm for ship localization in SAR images to balance the accuracy, speed, and computational cost. The proposed model reduces the computational complexity without compromising the accuracy. It follows a single-stage object detector algorithm having a novel backbone called S-Net, which is computationally less expensive and more accurate than the existing popular models. This demonstrates the practicality and robustness of the model. A combined loss function was also introduced for optimizing the model performance. An in-depth analysis of the quantitative and qualitative results of the proposed model was conducted in comparison with the existing state-of-the-art models. The results showed an improvement of 12.58 points in precision and 7.39 points in recall metrics for the proposed model with the proposed combined loss function over the Faster RCNN baseline model. A failure case scenario was also analyzed to examine the model's vulnerability. Furthermore, a sensitivity analysis was conducted for the dataset with various state-of-the-art models (including the proposed model) based on false alarm rate. It shows that the proposed model is the least sensitive for this dataset, indicating the model's robustness. The proposed algorithm can be applied to SAR images and optical remote sensing images (RGB) with a modification in the CNN architecture. This will significantly help improve maritime monitoring and detect other visual phenomena (like wakes) in the ocean.

ACKNOWLEDGMENT

The authors would like to thank Y. Wang et al., for providing the dataset. They also thank the anonymous reviewers for their valuable comments.

REFERENCES

[1] J. Li, C. Xu, H. Su, L. Gao, and T. Wang, "Deep learning for SAR ship detection: Past, present and future," *Remote Sens.*, vol. 14, no. 11, p. 2712, Jun. 2022, doi: 10.3390/rs14112712.

[2] F. C. Robey, D. R. Fuhrmann, E. J. Kelly, and R. Nitzberg, "A CFAR adaptive matched filter detector," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 28, no. 1, pp. 208–216, Jan. 1992.

[3] Y. Xu, W. Xiong, Y. Lv, and H. Liu, "A new method based on two-stage detection mechanism for detecting ships in high-resolution SAR images," in *Proc. MATEC Web Conf.*, vol. 128, 2017, p. 01014, doi: 10.1051/matecconf/201712801014.

[4] P. Iervolino and R. Guida, "A novel ship detector based on the generalized-likelihood ratio test for SAR imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 8, pp. 3616–3630, Aug. 2017, doi: 10.1109/JSTARS.2017.2692820.

[5] L. Xu, H. Zhang, C. Wang, B. Zhang, and S. Tian, "Compact polarimetric SAR ship detection with m-δ decomposition using visual attention model," *Remote Sens.*, vol. 8, no. 9, p. 751, Sep. 2016, doi: 10.3390/rs8090751.

[6] M.-D. Li, X.-C. Cui, and S.-W. Chen, "Adaptive superpixel-level CFAR detector for SAR inshore dense ship detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2021.3059253.

[7] G. Liu, X. Zhang, and J. Meng, "A small ship target detection method based on polarimetric SAR," *Remote Sens.*, vol. 11, no. 24, p. 2938, Dec. 2019, doi: 10.3390/rs11242938.

[8] A. Lupidi, D. Staglianò, M. Martorella, and F. Berizzi, "Fast detection of oil spills and ships using SAR images," *Remote Sens.*, vol. 9, no. 3, p. 230, Mar. 2017, doi: 10.3390/rs9030230.

[9] O. Karakus, I. Rizaev, and A. Achim, "Ship wake detection in SAR images via sparse regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 1665–1677, Mar. 2020, doi: 10.1109/TGRS.2019.2947360.

[10] J. Zhu, X. Qiu, Z. Pan, Y. Zhang, and B. Lei, "Projection shape template-based ship target recognition in TerraSAR-X images," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 2, pp. 222–226, Feb. 2017, doi: 10.1109/LGRS.2016.2635699.

[11] T. Zhang, X. Zhang, J. Shi, and S. Wei, "High-speed ship detection in SAR images by improved YOLOv3," in *Proc. 16th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process.*, Dec. 2019, pp. 149–152, doi: 10.1109/ICCWAMTIP47768.2019.9067695.

[12] T. Chengsheng, L. Huacheng, and X. Bing, "AdaBoost typical algorithm and its application research," in *Proc. MATEC Web Conf.*, vol. 139, Jan. 2017, p. 00222, doi: 10.1051/matecconf/201713900222.

[13] L. Rokach and O. Maimon, "Decision trees," in *The Data Mining and Knowledge Discovery Handbook*. Boston, MA, USA: Springer, 2005, ch. 9, pp. 165–192, doi: 10.1007/0-387-25465-X_9.

[14] T. Evgeniou and M. Pontil, *Support Vector Machines: Theory and Applications*, vol. 177. Berlin, Germany: Springer, 2005.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587, doi: 10.1109/CVPR.2014.81.

[16] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," 2015, *arXiv:1506.01497*.

[18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Computer Vision—ECCV*. Amsterdam, The Netherlands: Springer, Oct. 2016.

[19] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Miami, FL, USA, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.

[21] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.

[22] X. Xu, X. Zhang, and T. Zhang, "Lite-YOLOv5: A lightweight deep learning detector for on-board ship detection in large-scene Sentinel-1 SAR images," *Remote Sens.*, vol. 14, no. 4, p. 1018, Feb. 2022, doi: 10.3390/rs14041018.

[23] X. Wang, G. Li, A. Plaza, and Y. He, "Ship detection in SAR images by aggregating densities of Fisher vectors: Extension to a global perspective," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5206613, doi: 10.1109/TGRS.2021.3073053.

[24] X. Geng, L. Shi, J. Yang, P. Li, L. Zhao, W. Sun, and J. Zhao, "Ship detection and feature visualization analysis based on lightweight CNN in VH and VV polarization images," *Remote Sens.*, vol. 13, no. 6, p. 1184, Mar. 2021, doi: 10.3390/rs13061184.

[25] T. Miao, H. Zeng, W. Yang, B. Chu, F. Zou, W. Ren, and J. Chen, "An improved lightweight RetinaNet for ship detection in SAR images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4667–4679, 2022, doi: 10.1109/JSTARS.2022.3180159.

[26] B. Xiong, Z. Sun, J. Wang, X. Leng, and K. Ji, "A lightweight model for ship detection and recognition in complex-scene SAR images," *Remote Sens.*, vol. 14, no. 23, p. 6053, Nov. 2022, doi: 10.3390/rs14236053.

[27] Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "A SAR dataset of ship detection for deep learning under complex backgrounds," *Remote Sens.*, vol. 11, no. 7, p. 765, Mar. 2019, doi: 10.3390/rs11070765.

[28] T. Zhang, X. Zhang, J. Li, X. Xu, B. Wang, X. Zhan, Y. Xu, X. Ke, T. Zeng, H. Su, I. Ahmad, D. Pan, C. Liu, Y. Zhou, J. Shi, and S. Wei, "SAR ship detection dataset (SSDD): Official release and comprehensive data analysis," *Remote Sens.*, vol. 13, no. 18, p. 3690, Sep. 2021, doi: 10.3390/rs13183690.

[29] S. Wei, X. Zeng, Q. Qu, M. Wang, H. Su, and J. Shi, "HRSID: A high-resolution SAR images dataset for ship detection and instance segmentation," *IEEE Access*, vol. 8, pp. 120234–120254, 2020, doi: 10.1109/ACCESS.2020.3005861.

[30] S. Lei, D. Lu, X. Qiu, and C. Ding, "SRSDD-v1.0: A high-resolution SAR rotation ship detection dataset," *Remote Sens.*, vol. 13, no. 24, p. 5104, Dec. 2021, doi: 10.3390/rs13245104.

[31] T. Zhang, X. Zhang, X. Ke, X. Zhan, J. Shi, S. Wei, D. Pan, J. Li, H. Su, Y. Zhou, and D. Kumar, "LS-SSDD-v1.0: A deep learning dataset dedicated to small ship detection from large-scale Sentinel-1 SAR images," *Remote Sens.*, vol. 12, no. 18, p. 2997, Sep. 2020, doi: 10.3390/rs12182997.

SHOVAKAR BHATTACHARJEE received the master's degree from Jadavpur University, Kolkata, West Bengal, India, in 2020. He is currently pursuing the Ph.D. degree with the Interdisciplinary Department of Ocean Engineering and the Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai, India. He holds a prestigious Prime Minister Research Fellowship (PMRF) for the Ph.D. research work. His current research interests include computer vision, microwave satellite image processing, and remote sensing.

PALANISAMY SHANMUGAM received the Ph.D. degree in optical/microwave remote sensing techniques from Anna University, Chennai, India, in 2002. He is currently a Professor with the Department of Ocean Engineering, Indian Institute of Technology Madras, Chennai. He has been a principal investigator of several projects funded by the Government of India. His current research interests include ocean optics and imaging, satellite oceanography, radiative transfer in the ocean, algorithm/model development, and underwater wireless optical communication.

SUKHENDU DAS (Senior Member, IEEE) received the Ph.D. degree from IIT Kharagpur, in 1993. Since 1989, he has been a Faculty with IIT Madras, Chennai, India, where he is currently a Professor with the Department of Computer Science and Engineering. His current research interests include visual perception, computer vision: digital image processing and pattern recognition, computer graphics, artificial neural networks, computational science and engineering, soft computing, deep learning, and computational brain modeling.