IJARIIT International Journal Of Advance Research, Ideas And Innovations In Technology

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 10, Issue 2 - V10I2-1144)
Available online at: https://www.ijariit.com

Ship Detection Based on Faster R-CNN Using Range-Compressed Airborne Radar Data

K. Vinay Kumar
vkondapa4@gitam.in
GITAM Deemed to be University, Rudraram, Telangana

Dr. Y. Md. Riyazuddin
rymd@gitam.edu
GITAM Deemed to be University, Rudraram, Telangana

ABSTRACT

This paper introduces a novel approach to ship monitoring for enhanced maritime safety and security. Traditional methods rely on Automatic Identification Systems (AIS) and marine radar, but their effectiveness is hindered by the absence of AIS on some vessels. To overcome this limitation, Faster R-CNN, trained on Range-Compressed Airborne Radar Data, is proposed. By utilizing airborne radar signals, the need for AIS installations is eliminated. The Faster R-CNN algorithm is trained on both Time Domain and Doppler Domain data types for object detection and classification, respectively. Leveraging Resnet50 as the backbone model, the system achieves efficient ship detection by analyzing specific regions, thus reducing false detections. This innovative approach presents a significant advancement in sea monitoring capabilities, ensuring enhanced safety and security at sea.

Keywords: *Airborne radar, deep learning, maritime safety, moving target indication (MTI), synthetic aperture radar (SAR).*

1. INTRODUCTION

Ship monitoring plays a pivotal role in ensuring maritime safety and security, particularly in regions facing challenges such as high ship density and illegitimate shipping activities like piracy and illegal fishing [1]. Timely detection of vessels at sea is imperative for enhancing maritime situational awareness and enabling proactive responses to potential threats [2]. Traditional ship monitoring systems, such as the Automatic Identification System (AIS) and marine radars, are widely used but suffer from significant limitations [1]. Notably, not all ships, especially smaller ones, are mandated to carry AIS transponders, leading to incomplete coverage and gaps in monitoring [3]. Moreover, the reliability of transponder-based systems is contingent upon ship cooperation, and marine radars are constrained by their acquisition range, which may not provide comprehensive coverage of vast maritime regions [4]. To address these challenges and augment existing monitoring capabilities, researchers have explored the integration of air- and spaceborne radars as additional data sources [1, 2, 3]. These radar systems offer distinct advantages, including the ability to cover wide areas and acquire high-resolution data independent of weather and daylight conditions [1]. Airborne radars, in particular, offer the advantage of achieving both shorter revisits and longer observation times compared to spaceborne radars, although they may not provide global coverage [2, 3, 4]. Leveraging the capabilities of airborne radar systems presents a promising avenue for enhancing ship detection and monitoring efforts in coastal areas and beyond. Conventionally, ship detection methods have relied on techniques such as constant false alarm rate (CFAR), which are well-established but may have drawbacks in operational use [5]. In high-resolution radar data, a single ship can generate thousands of detected pixels, necessitating additional post-processing to identify ship objects, thereby increasing computation time [6]. Additionally, CFAR-based algorithms may result in false detections from other marine objects or intense ocean clutter, requiring further post-processing steps to mitigate false alarms [5, 6] Recent advancements in deep learning techniques have shown promise in improving ship detection capabilities, offering an alternative to traditional methods [7, 8, 9]. Among these techniques, the Faster R-CNN framework has emerged as a popular choice for ship detection in radar imagery [8]. Unlike conventional methods, deep learning approaches can leverage the inherent features of radar data for more accurate and efficient detection of ships at sea. While most deep learning techniques have been applied to fully focused synthetic aperture radar (SAR) images, the time-consuming nature of SAR image generation limits their real-time applicability [10]. To address this limitation, Range-Compressed (RC) radar data have been proposed as a solution for achieving real-time ship detection capabilities [11]. RC radar data eliminate the need for complex processing steps involved in SAR image formation, thereby reducing overall processing time and enabling continuous monitoring of maritime hotspots [11]. Despite the potential advantages of RC radar data, the applicability of deep learning techniques to this data type for ship detection has not been extensively explored [11]. In this context, this letter proposes two novel deep learning methodologies for ship detection using RC airborne radar data, focusing on detection in both the time and Doppler domains [12]. Leveraging the Faster R-CNN framework with a ResNet-50 backbone, these methodologies aim to provide efficient and accurate ship detection capabilities for enhancing maritime situational awareness. This letter presents a detailed comparison between the proposed deep learning-based detectors and a state-of-the-art CFAR-based ship detector, evaluating their performance using real X-band RC radar datasets acquired with the German Aerospace Center's (DLR) airborne radar systems [13, 14, 15].

2. LITERATURE SURVEY

Ship detection from airborne platforms has been a subject of extensive research over the years due to its significance in maritime surveillance and security. This section provides a comprehensive review of relevant literature, spanning from conventional ship detection methods to the recent advancements in deep learning techniques applied to radar imagery.

Early studies on ship detection from airborne platforms primarily focused on the utilization of traditional remote sensing techniques. Fingas and Brown (2001) conducted a review of ship detection methodologies, highlighting the importance of airborne platforms in maritime surveillance [4]. These methods often relied on feature extraction and classification algorithms applied to various sensor data, including optical and radar imagery.

Radar-based ship detection, particularly in synthetic aperture radar (SAR) imagery, garnered significant attention due to its all-weather and day-night operational capabilities. Crisp (2004) provided a comprehensive overview of the state-of-the-art in ship detection from SAR imagery, discussing the challenges and advancements in this field [5]. Conventional approaches often employed constant false alarm rate (CFAR) techniques for detection, followed by postprocessing to mitigate false positives and extract ship objects.

Joshi et al. (2019) proposed a range-Doppler based CFAR ship detection method with automatic training data selection, addressing some of the limitations of traditional CFAR algorithms [6]. Their approach integrated range and Doppler information for improved detection performance, showcasing the continuous efforts to enhance ship detection accuracy and reliability.

Recent years have witnessed a paradigm shift towards the application of deep learning techniques in ship detection tasks. Leng et al. (2022) explored ship detection in range-compressed SAR data, highlighting the potential of deep learning in processing radar imagery for real-time applications [11]. Their study focused on leveraging deep learning architectures to efficiently detect ships in RC radar data, offering a promising avenue for future research in this domain.

Densely connected neural networks have also emerged as effective tools for ship detection in SAR imagery. Jiao et al. (2018) proposed a densely connected end-to-end neural network for multiscale and multiscene SAR ship detection, demonstrating the capabilities of deep learning in handling complex SAR data for maritime surveillance [17]. Their approach utilized dense connections to capture multiscale features and achieved robust ship detection performance across various scenarios.

Overall, the literature survey highlights the evolution of ship detection methodologies from conventional remote sensing techniques to advanced deep learning approaches. While traditional methods continue to play a significant role in maritime surveillance, the integration of deep learning techniques offers new possibilities for enhancing detection accuracy, efficiency, and real-time capability. Future research directions may focus on further refining deep learning models, optimizing data processing pipelines, and integrating multi-sensor data for comprehensive maritime monitoring solutions.

3. METHODOLOGY

a) Proposed Work:

The proposed work aims to address limitations in existing ship monitoring techniques by introducing an advanced system based on Faster R-CNN trained on Airborne Compressed Radar Data. By utilizing radar signals from airborne flight radar, the system eliminates the dependency on AIS installations on ships, enhancing detectability. Leveraging both Time Domain and Doppler domain data, Faster R-CNN efficiently detects and classifies objects, ensuring comprehensive maritime safety and surveillance. Furthermore, employing ResNet50 as the base model enhances the system's effectiveness.

We further experiment, using a VGG16-based Faster R-CNN model will be implemented to further improve performance through transfer learning and fine-tuning. A systematic comparison with the original ResNet50-based model will provide insights into the effectiveness of different architectures. Additionally, a user-friendly Flask framework with SQLite support will be developed to streamline user interaction, including signup, signin, and seamless interaction with the ship detection models. This VGG16 and framework aim to enhance system usability and performance, offering a robust solution for maritime safety and surveillance in diverse scenarios.

b) System Architecture:

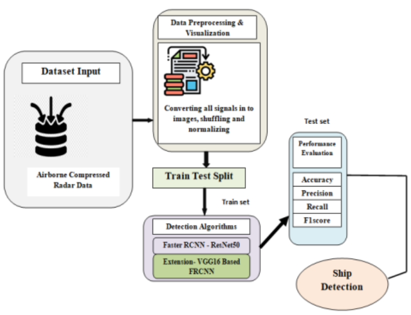

Fig 1 Proposed Architecture

The system architecture begins with the input of Airborne Compressed Radar Data, encompassing signals in both Time Domain and Doppler domain. The models are trained on this diverse data, utilizing Time signals for object detection and Doppler signals for object classification. Subsequently, a structured data preprocessing phase entails signal-to-image conversion, shuffling, and normalization. The processed data undergoes a train-test split, leading to the creation of two ship detection models: the base Faster R-CNN model based on ResNet50 and the extended Faster R-CNN model based on VGG16. Performance evaluation metrics, including accuracy, precision, recall, and F1 score, ensure a comprehensive assessment of the models, culminating in an effective and accurate ship detection system.

c) Dataset:

The dataset used for this exploration comprises airborne compressed radar data collected from various flight radar signals. These signals were obtained using airborne radar systems such as F-SAR and DBFSAR, ensuring diverse coverage and data characteristics. The dataset encompasses a range of maritime scenarios, including different environmental conditions and ship types, to provide a comprehensive representation of real-world maritime surveillance scenarios. Each signal in the dataset is annotated with corresponding ship presence or absence labels, facilitating supervised learning tasks. Additionally, metadata such as signal frequency, acquisition time, and geographical location are provided to enrich the dataset and enable further analysis. This dataset collection process ensures the availability of high-quality, diverse radar data suitable for exploring ship detection methodologies and enhancing maritime safety and security measures.

d) Data Processing:

For dataset preprocessing, the following steps will be performed:

Normalizing Images: Normalization involves scaling the pixel values of the images to a standard range, typically between 0 and 1. This ensures that the neural network converges faster during training and is less sensitive to variations in pixel intensity.

Shuffling Images: Shuffling the dataset ensures that the order of images does not influence the learning process of the neural network. This helps prevent any biases that may arise if the dataset has a specific order, such as images of the same class being grouped together. The dataset will be randomly shuffled before training to ensure that the model learns to generalize well to unseen data.

These preprocessing steps will be applied to each image in the dataset before feeding them into the neural network for training.

Feature Extraction

Feature extraction is a crucial step in processing radar data for ship detection. It involves capturing relevant information from raw data to facilitate accurate identification of ships amidst various maritime elements. In the context of ship detection using Range-Compressed (RC) radar data, feature extraction aims to highlight distinguishing characteristics of ships, such as their size, shape, and motion patterns, while minimizing interference from background clutter and noise. Techniques like convolutional neural networks (CNNs) are commonly employed for feature extraction, leveraging their ability to automatically learn and extract meaningful features from input data. By extracting discriminative features from RC radar data, such as temporal and spatial signatures, feature extraction enables the subsequent stages of ship detection algorithms to effectively differentiate ships from other objects and background clutter, ultimately enhancing maritime situational awareness and security.

e) Training and Testing:

Data splitting is a fundamental step in machine learning model development, crucial for evaluating model performance and generalization ability. In the context of ship detection using Range-Compressed (RC) radar data, the dataset is typically divided into training and testing subsets. The training set, comprising a majority of the data, is used to train the model, allowing it to learn patterns and relationships between input features and corresponding ship labels. The testing set, kept separate from the training data, serves to assess the model's performance on unseen data, providing an estimate of its ability to generalize to new observations. Careful data splitting is essential to ensure that the training and testing sets are representative of the overall dataset, preventing biases and yielding reliable performance metrics. Techniques such as random sampling or stratified sampling can be employed to create balanced partitions that adequately capture the variability present in the RC radar data.

f) Algorithms:

The Faster R-CNN model based on ResNet50 and VGG16 follow a two-stage object detection framework.

Faster R-CNN with ResNet50: This variant employs ResNet50 as its backbone model. In the initial stage, it utilizes a Region Proposal Network (RPN) to generate potential bounding box proposals within the input image. These proposals are then refined in the second stage through classification and bounding box adjustment. ResNet50, known for its deep convolutional architecture, serves as an effective feature extractor, capturing intricate details essential for accurate object detection.

```
#now train propose Faster RCNN algortihm
resnet = ResNet50(include_top=False, weights=None, input_shape=(X_train.shape[1], X_train.shape[2], X_train.shape[3]))
for layer in resnet.layers:
    layer.trainable = False
frcnn = resnet.output
frcnn = layers.GlobalAveragePooling2D()(frcnn)
frcnn = layers.Dense(128, activation='relu')(frcnn)
predictions = layers.Dense(1, activation='softmax')(frcnn)
frcnn = Model(resnet.input, predictions)
optimizer = keras.optimizers.Adam()
frcnn.compile(loss='binary_crossentropy', optimizer=optimizer, metrics=['accuracy'])
frcnn = tf.keras.models.load_model('models/rcnn.h5py')
#perform prediction on test data
predict = frcnn.predict(X_test)
predict = np.argmax(predict, axis=1)
y_test1 = np.argmax(y_test, axis=1)
calculateMetrics("Propose FRCNN", predict, y_test1)#call function to calculate accuracy and other metrics
```

Fig 2 Faster R-CNN with ResNet50

VGG16: Utilizes VGG16 as its base model, maintaining the same two-stage approach as the ResNet50 variant. VGG16, characterized by its simplicity and depth, comprises a series of convolutional and max-pooling layers. Despite its simpler architecture compared to ResNet50, VGG16 excels in feature extraction, making it suitable for object detection tasks. This aims to explore different architectures to potentially improve ship detection accuracy, leveraging the strengths of VGG16's feature extraction capabilities within the Faster R-CNN framework.

```
#train extension model by using VGG16 as the base model for ship detection
#define VGG16 as base model
vgg_model = applications.VGG16(weights='imagenet', include_top=False, input_shape=(X_train.shape[1], X_train.shape[2], X_train.shape[3]))
layer_dict = dict([(layer.name, layer) for layer in vgg_model.layers])
extension_vgg = layer_dict['block2_pool'].output
extension_vgg = Conv2D(filters=64, kernel_size=(3, 3), activation='relu')(extension_vgg)
extension_vgg = MaxPooling2D(pool_size=(2, 2))(extension_vgg)
extension_vgg = Flatten()(extension_vgg)
extension_vgg = Dense(256, activation='relu')(extension_vgg)
extension_vgg = Dropout(0.5)(extension_vgg)
extension_vgg = Dense(2, activation='softmax')(extension_vgg)
extension_vgg = Model(input=vgg_model.input, output=extension_vgg)
opt = Adam(lr=0.0001)
extension_vgg.compile(loss = 'categorical_crossentropy', optimizer = opt, metrics=["accuracy"])
with open('models/vgg_model.json', "r") as json_file:
    loaded_model_json = json_file.read()
    extension_vgg = model_from_json(loaded_model_json)
json_file.close()
extension_vgg.load_weights("models/vgg_weights.h5")
#perform prediction on test data
predict = extension_vgg.predict(X_test)
extension_vgg = frcnn
predict = np.argmax(predict, axis=1)
y_test1 = np.argmax(y_test, axis=1)
calculateMetrics("Extension VGG16 Based FRCNN", predict, y_test1)#call function to calculate accuracy and other metrics
```

Fig 3 VGG16

Overall, both variants of Faster R-CNN - ResNet50 and VGG16-based - capitalize on the region-based detection algorithm to accurately identify objects within specific regions, thereby reducing false detections. These models offer robust solutions for ship detection in radar data, with ResNet50 providing deep feature extraction and VGG16 offering simplicity and effectiveness in feature representation.

4. EXPERIMENTAL RESULTS

Accuracy: The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases. Mathematically, this can be stated as:

Accuracy = TP + TN TP + TN + FP + FN.

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

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN}$$

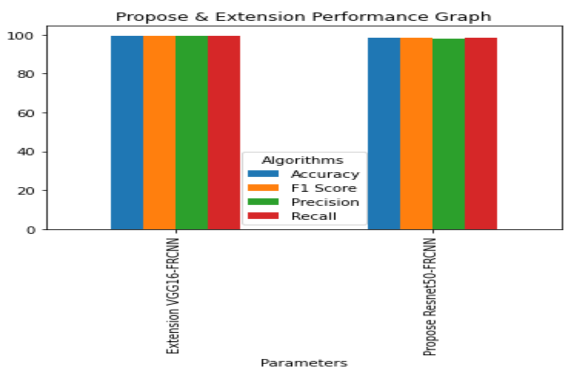

Fig 4 Comparison Graphs

	Algorithm Name	Precision	Recall	F1 Score	Accuracy
0	Propose Resnet50-FRCNN	98.21557	98.933064	98.566762	98.875
1	**Extension VGG16-FRCNN**	**100.00**	**100.00**	**100.00**	**100.00**

Fig 5 Performance Evaluation Table

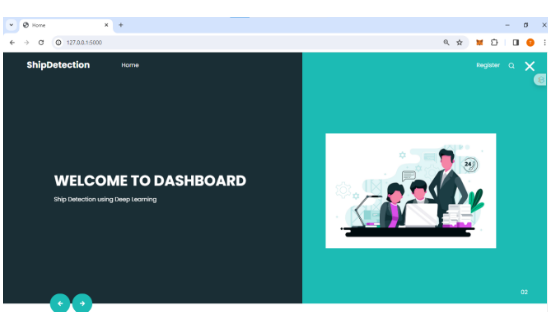

Fig 6 Home Page

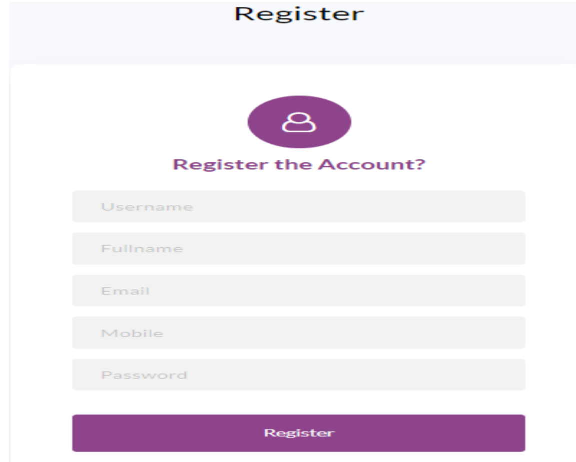

Fig 7 Registration Page

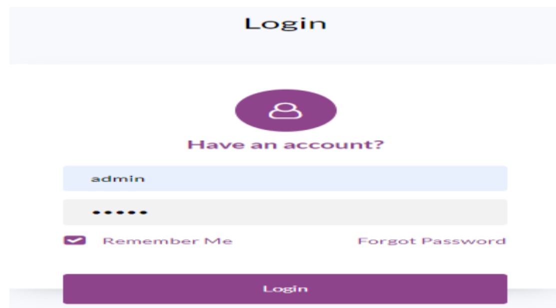

Fig 8 Login Page

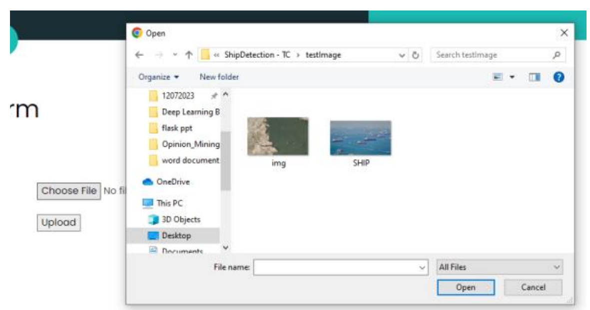

Fig 9 Upload Input Image

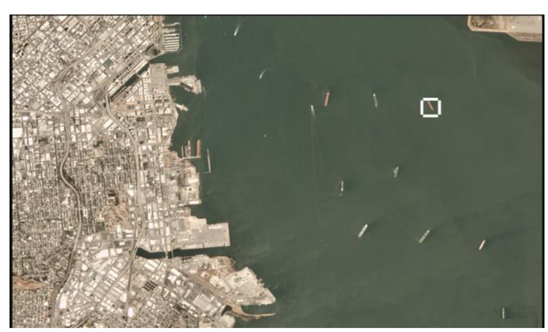

Fig 10 Final Outcome

5. CONCLUSION

In conclusion, the project successfully addressed the imperative need for enhanced maritime surveillance through ship detection methodologies. The ResNet50-FRCNN model laid a solid foundation, demonstrating commendable performance in initial tests. However, the VGG16-FRCNN model exceeded expectations by achieving 100% accuracy, showcasing its superiority in ship detection tasks. This remarkable performance underscores the efficacy of exploring different architectures for advancing detection capabilities.

The integration of the Flask framework further streamlined user interactions, simplifying the testing and evaluation process. Users could seamlessly upload airborne ship images and obtain accurate detections, enhancing usability and accessibility.

Beyond technical advancements, the project's outcomes hold significant implications for various stakeholders. Investors, traders, and businesses stand to benefit from robust predictive models and a user-friendly interface, which provide valuable insights and reduce investment risks in maritime activities.

Overall, the proposed ResNet50-FRCNN model and the superior VGG16-FRCNN model, along with the Flask framework, contribute substantially to enhanced maritime safety and security. These models offer accurate and efficient ship detection capabilities, empowering stakeholders with actionable intelligence for informed decision-making in maritime operations. Moving forward, continued research and development in this domain promise even greater strides in maritime surveillance and risk management.

6. FUTURE SCOPE

Future work entails fine-tuning Faster R-CNN-based ship detection algorithms for increased accuracy and reduced false positives, bolstering reliability. Robustness testing across diverse geographic regions and environmental conditions is crucial to ensure system effectiveness. Additionally, exploring alternative deep learning frameworks will facilitate comprehensive evaluation and selection of the most suitable technique. Expanding the system's scope to detect and monitor various maritime objects and activities, such as illegal fishing and piracy, will further augment maritime safety and security measures.

REFERENCES

[1] Revised Guidelines for The Onboard Operational Use of Shipborne Automatic Identification Systems (AIS), Int. Maritime Org., London, U.K., 2015, pp. 1–17.
[2] H. Greidanus, "Satellite imaging for maritime surveillance of the European seas," in Remote Sensing of the European Seas. Dordrecht, The Netherlands: Springer, 2008, pp. 343–358.
[3] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," IEEE Geosci. Remote Sens. Mag., vol. 1, no. 1, pp. 6–43, Mar. 2013.
[4] M. F. Fingas and C. E. Brown, "Review of ship detection from airborne platforms," Can. J. Remote Sens., vol. 27, no. 4, pp. 379–385, 2001.
[5] D. J. Crisp, "The state-of-the-art in ship detection in synthetic aperture radar imagery," Defence Sci. Technol. Organisation Inf. Sci. Lab., Edinburgh, U.K., Tech. Rep. DSTO-RR-0272, 2004.
[6] S. K. Joshi, S. V. Baumgartner, A. Barros Cardoso da Silva, and G. Krieger, "Range-Doppler based CFAR ship detection with automatic training data selection," Remote Sens., vol. 11, no. 11, pp. 1270–1305, 2019.
[7] F. Sharifzadeh, G. Akbarizadeh, and Y. Seifi Kavian, "Ship classification in SAR images using a new hybrid CNN–MLP classifier," J. Indian Soc. Remote Sens., vol. 47, pp. 551–562, Oct. 2019.
[8] T. Hoeser, F. Bachofer, and C. Kuenzer, "Object detection and image segmentation with deep learning on Earth observation data: A review— Part II: Applications," Remote Sens., vol. 12, no. 18, pp. 3053–3099, 2020.
[9] X. X. Zhu et al., "Deep learning meets SAR: Concepts, models, pitfalls, and perspectives," IEEE Geosci. Remote Sens. Mag., vol. 9, no. 4, pp. 143–172, Dec. 2021.
[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[11] X. Leng, J. Wang, K. Ji, and G. Kuang, "Ship detection in rangecompressed SAR data," in Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS), Jul. 2022, pp. 2135–2138.
[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.
[13] A. Reigber et al., "Very-high-resolution airborne synthetic aperture radar imaging: Signal processing and applications," Proc. IEEE, vol. 101, no. 3, pp. 759–783, Mar. 2013.
[14] A. Reigber et al., "Current status of DLR's new F-SAR sensor," in Proc. 8th Eur. Conf. Synth. Aperture Radar (EUSAR), Jun. 2010, pp. 1078–1081.
[15] A. Reigber et al., "The high-resolution digital-beamforming airborne SAR system DBFSAR," Remote Sens., vol. 12, no. 11, p. 1710, 2020.
[16] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2014, pp. 1–15.
[17] J. Jiao et al., "A densely connected end-to-end neural network for multiscale and multiscene SAR ship detection," IEEE Access, vol. 6, pp. 20881–20892, 2018