SHIP DETECTION BASED ON FASTER R-CNN USING RANGE-COMPRESSED AIRBORNE RADAR DATA

Nagireddy Pradeep[1], Dr S.Srinivas[2]
[1]*PG Scholar, Department of CSE, Holy Mary Institute of Technology and Science, Hyderabad*
pradeepnagireddy.it@gmail.com
[2]*Assoc. Professor, Department of CSE, Holy Mary Institute of Technology and Science, Hyderabad*
srinivas.s@hmgi.ac.in

ABSTRACT

Near real-time ship surveillance is critical for maritime safety and security. Automatic identification systems (AIS) and marine radars are two well-established ship monitoring systems. However, not all ships are required to have AIS transponders, and marine radars have restricted visibility. For these reasons, airborne radars can be utilized as an extra and supporting sensor for ship surveillance, particularly on broad waters. Modern ship identification algorithms in radar imaging use a constant false alarm rate (CFAR). Because such algorithms are pixel-based, near-real-time detection can be difficult to perform in practice. This letter introduces two object-oriented ship detectors based on the faster R-CNN. The first detector reads airborne Range-Compressed (RC) radar data patches in time, while the second reads them in Doppler domain. The Faster R-CNN models are trained on thousands of genuine X-band aerial RC radar data patches with multiple ship signals. The suggested object-oriented ship detectors are evaluated on numerous scenarios, and the models demonstrate strong recall performance even in very dense multitarget scenarios in the North Sea's complicated inshore environment.

I. INTRODUCTION

In summary, this letter addresses the need of near real-time ship monitoring in ensuring maritime safety and security, particularly in settings where conventional systems such as AIS and marine radars have limits. While AIS may not be widely used, and marine radars may have limited visibility, the incorporation of airborne radars emerges as an important additional sensor, particularly in open sea situations. The typical constant false alarm rate (CFAR) algorithms for ship recognition in radar imaging, which are pixel-based, face obstacles in attaining near-real-time detection. This letter offers two new object-oriented ship detectors based on the Faster Region-Based Convolutional Neural Network (R-CNN) paradigm. The first detector functions in the time domain, whilst the second operates in the Doppler domain of airborne Range-Compressed (RC) radar data patches. These Faster R-CNN models perform well after being trained on large datasets of genuine X-band aerial RC radar data patches with various ship signals. The suggested object-oriented ship detectors have been evaluated in a variety of scenarios, including dense multitarget situations in the complicated inshore environment of the North Sea. This novel approach represents a substantial development in near-real-time ship monitoring, providing improved accuracy and reliability under difficult marine situations.

OBJECTIVE

The primary objectives of this project are to develop and implement two distinct ship detection algorithms based on Faster Region-Based Convolutional Neural Networks (R-CNNs). One detector will operate in the time domain, while the other will focus on the Doppler domain of airborne Range-Compressed (RC) radar data patches. The project aims to train these models on a comprehensive dataset comprising real X-band airborne RC radar data patches, ensuring the robustness and generalization of the ship detectors across diverse scenarios and environmental conditions. Optimization efforts will be directed towards achieving near real-time capabilities, overcoming challenges associated with constant false alarm rate (CFAR) algorithms for timely and responsive ship monitoring. Comprehensive testing and evaluation will be conducted in various scenarios, including dense multitarget situations in complex inshore environments like the North Sea, to validate the effectiveness and reliability of the proposed ship detectors. Integration with existing systems, such as the Automatic Identification System (AIS) and marine radars, will be explored to enhance the overall monitoring infrastructure. The project will prioritize the documentation of the development process, methodologies, and outcomes, with user-friendly documentation for potential deployment.

PROBLEM STATEMENT

Effective and timely ship identification is the main difficulty in maritime surveillance, especially in situations where Automatic Identification Systems (AIS) are unavailable or where visibility is poor because of the limited range of marine radars. Near-real-time detection in dense multitarget environments is a challenge for traditional pixel-based algorithms, such as those that employ a Constant False Alarm Rate (CFAR). Two object-oriented ship detectors based on the Faster R-CNN architecture are proposed in this paper to overcome these issues. Range-Compressed (RC) radar data patches in the air are processed immediately in time by the first detector and in the Doppler domain by the second. These detectors are intended to increase the accuracy of ship identification in challenging marine conditions, such the North Sea's inshore areas, where it might be difficult to keep an eye on several ships.

EXISTING SYSTEM

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

Disadvantage of Existing System

- Less accuracy
- Difficult to predict

PROPOSED SYSTEM

Region-Based Convolutional Neural Networks (R-CNN) and PyTorch segmentation are two significant concepts in computer vision and deep learning. R-CNN is a family of object detection models that localize and classify objects within an image. The original R-CNN approach involves two main steps: region proposal and object classification. It first generates potential object regions in an image using a selective search algorithm. Then, these proposed regions are forwarded through a CNN for feature extraction, and a classifier is applied to each region to determine the object category. While effective, this approach is computationally expensive due to its two-step process. The subsequent Faster R-CNN and Mask R-CNN architectures address this issue by introducing region proposal networks (RPNs) and enabling end-to-end training. Faster R-CNN efficiently integrates the region proposal and object detection tasks, while Mask R-CNN extends it further to include instance segmentation, providing pixel-level object delineation. PyTorch, a popular deep learning framework, provides tools and modules for implementing segmentation tasks. Segmentation involves dividing an image into meaningful regions or segments, often corresponding to objects or areas of interest. PyTorch's torchvision library offers pre-trained models for semantic segmentation, such as FCN (Fully Convolutional Network), UNet, and DeepLabV3, allowing users to leverage these models for various segmentation applications. PyTorch also supports the implementation of custom segmentation models using its flexible neural network architecture. In the context of PyTorch, segmentation tasks often involve the use of convolutional neural networks (CNNs) with encoder-decoder architectures. The encoder extracts hierarchical features from the input image, and the decoder reconstructs the segmentation map. U-Net, for example, is a popular architecture for semantic segmentation in PyTorch, known for its skip connections that aid in preserving spatial information.

Both R-CNN and PyTorch segmentation contribute significantly to computer vision tasks. R-CNN is particularly powerful for object detection, while PyTorch's segmentation capabilities are versatile and applicable to a range of image segmentation tasks. Depending on the specific requirements of a project, one might choose between these methodologies to achieve the desired outcome in terms of object localization, classification, or pixel-wise segmentation.

Advantages of Proposed System

- accurate object localization
- Robust to Object Occlusion
- Easy to predict

RELATED WORKS

Automatic Identification Systems (AIS) and maritime radars have been the main subjects of earlier ship monitoring research. Although AIS offers useful information such as ship identity, position, and speed, its coverage is limited since it is not required for all vessels. Despite their widespread usage, marine radars have limited visibility, especially in locations with high traffic or unfavorable weather. These drawbacks emphasize the necessity of investigating other sensor systems, including aerial radars, in order to improve surveillance capabilities. A common technique in radar signal processing, CFAR algorithms are made to differentiate between real targets and noise by keeping the false alarm rate constant. However, because of their processing complexity and limited capacity to handle dense multitarget situations, these pixel-based approaches may have trouble with near-real-time detection. Numerous research have improved ship recognition in radar data by utilizing machine learning and deep learning approaches. Ship identification and tracking have been automated using techniques like convolutional neural networks (CNNs) and their variations, including Faster R-CNN. Deep learning has demonstrated the potential to greatly increase detection accuracy over conventional techniques, particularly in congested settings. Nevertheless, a research gap still exists in the application of these methods to aerial Range-Compressed (RC) radar data in intricate marine situations. Faster R-CNN models' capacity to effectively handle overlapping targets and detect objects at various sizes has drawn interest in its application in maritime surveillance. Only conventional X-band radar data or satellite photos have been used in the past. Measures including recall, precision, and F1-score were commonly used in previous studies to assess detection ability; however, these analyses were frequently restricted to single-target situations or more straightforward settings. This paper offers important insights into how Faster R-CNN may be modified and refined for near-real-time ship identification across a range of marine situations by extending the performance evaluation to more complicated scenarios with numerous targets. The models' potential to improve marine safety and security is shown by their great recall performance, even in scenarios with a large number of targets.

METHODOLOGY OF PROJECT

Two object-oriented ship identification techniques based on the Faster R-CNN architecture are presented in this paper. The first technique captures spatial characteristics in the picture domain by processing airborne Range-Compressed (RC) radar data patches immediately in time. Doppler data is the subject of the second approach, which interprets the frequency shifts in the radar signal that correlate to ship movement. Both techniques take use of Faster R-CNN's multi-scale object detection capabilities and effective handling of overlapping targets. A vast collection of X-band aerial RC radar data patches with many ship signals is used to train the models. The models' recall ability is evaluated in a variety of circumstances, including dense multitarget contexts. These approaches seek to increase the precision and dependability of ship identification, particularly in demanding marine environments.

MODULE DESCRIPTION:

Data Collection:

Gathering pertinent data for your machine learning assignment is the first stage in this process. Your model's performance is greatly influenced by the caliber and volume of your data. This might include manually gathering and annotating data or obtaining datasets from a variety of sources, including databases and APIs.

Investigating Data:

Exploring and comprehending your dataset is the next step after obtaining it. To learn more about how characteristics are distributed, spot trends, and comprehend connections within the data, statistical analysis, visualization, and summarization are used. Making educated choices on feature engineering and preprocessing is aided by exploratory data analysis, or EDA.

Falsification of Data:

The data is ready for model training in this stage. Cleaning the data entails addressing any discrepancies, anomalies, and missing numbers. Feature engineering can be used to change or add new features. To guarantee that every feature contributes equally to the model, data normalization and scaling may be used.\

Data Modeling:

On the basis of your goal (classification, regression, clustering, etc.), you select a machine learning model at this crucial stage. To train your model and test it, you separate your dataset into training and testing sets. The modeling process involves choosing algorithms, adjusting hyperparameters, and assessing the model's performance using metrics such as recall, accuracy, precision, and others depending on the job at hand.

Adaptability:

Training the chosen machine learning model on the training data is known as Adaptability. The model adapts its parameters to produce precise predictions after discovering patterns and correlations in the data. Following training, the model's performance on the test set is evaluated to see how well it generalizes to new data. The model's performance may be enhanced iteratively, and the finished product is prepared for deployment.

ALGORITHM USED IN PROJECT

Region-Based Convolutional Neural Networks (R-CNN) and PyTorch segmentation are two significant concepts in computer vision and deep learning.

R-CNN is a family of object detection models that localize and classify objects within an image. The original R-CNN approach involves two main steps: region proposal and object classification. It first generates potential object regions in an image using a selective search algorithm. Then, these proposed regions are forwarded through a CNN for feature extraction, and a classifier is applied to each region to determine the object category. While effective, this approach is computationally expensive due to its two-step process. The subsequent Faster R-CNN and Mask R-CNN architectures address this issue by introducing region proposal networks (RPNs) and enabling end-to-end training. Faster R-CNN efficiently integrates the region proposal and object detection tasks, while Mask R-CNN extends it further to include instance segmentation, providing pixel-level object delineation. PyTorch, a popular deep learning framework, provides tools and modules for implementing segmentation tasks. Segmentation involves dividing an image into meaningful regions or segments, often corresponding to objects or areas of interest. PyTorch's torch vision library offers pre-trained models for semantic segmentation, such as FCN (Fully Convolutional Network), UNet, and DeepLabV3, allowing users to leverage these models for various segmentation applications. PyTorch also supports the implementation of custom segmentation models using its flexible neural network architecture.

In the context of PyTorch, segmentation tasks often involve the use of convolutional neural networks (CNNs) with encoder-decoder architectures. The encoder extracts

hierarchical features from the input image, and the decoder reconstructs the segmentation map. U-Net, for example, is a popular architecture for semantic segmentation in PyTorch, known for its skip connections that aid in preserving spatial information.
Both R-CNN and PyTorch segmentation contribute significantly to computer vision tasks. R-CNN is particularly powerful for object detection, while PyTorch's segmentation capabilities are versatile and applicable to a range of image segmentation tasks. Depending on the specific requirements of a project, one might choose between these methodologies to achieve the desired outcome in terms of object localization, classification, or pixel-wise segmentation.

DATA FLOW DIAGRAM

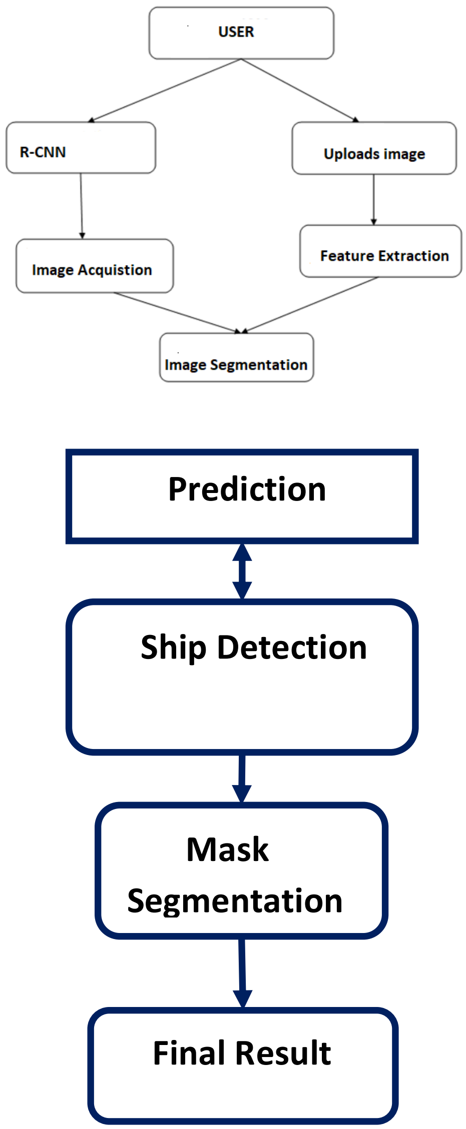


Fig: 6 Flow Diagram

SYSTEM ARCHITECTURE

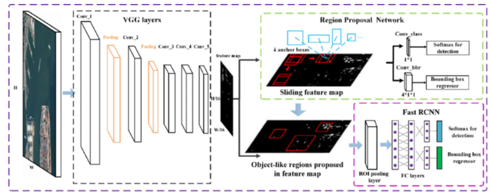
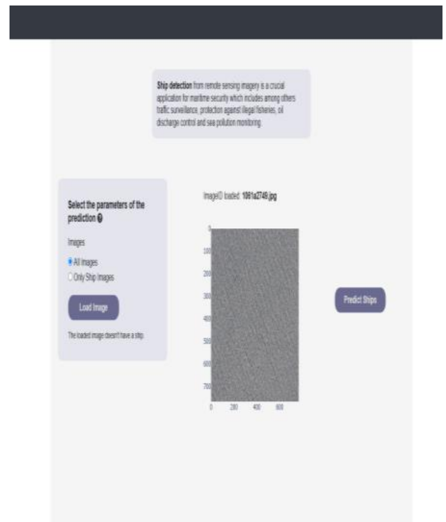
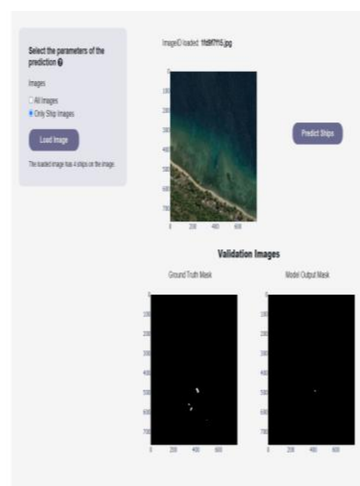
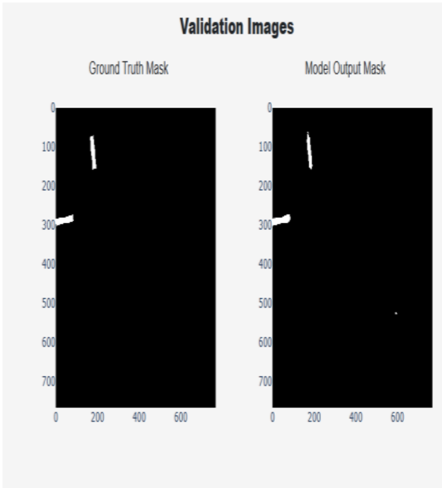


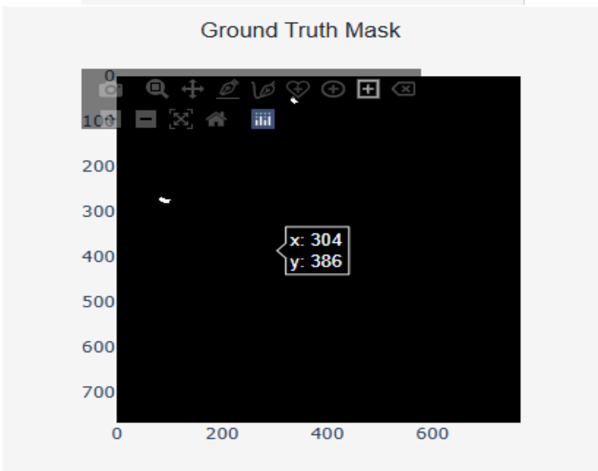
Fig: 7 System Architecture Of Project

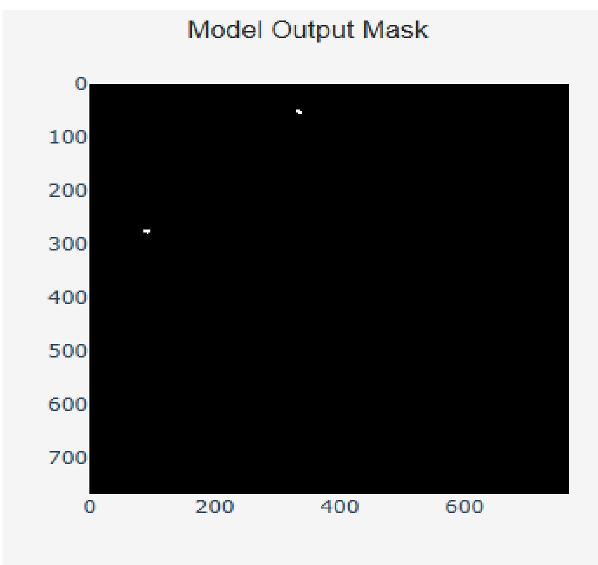
RESULTS

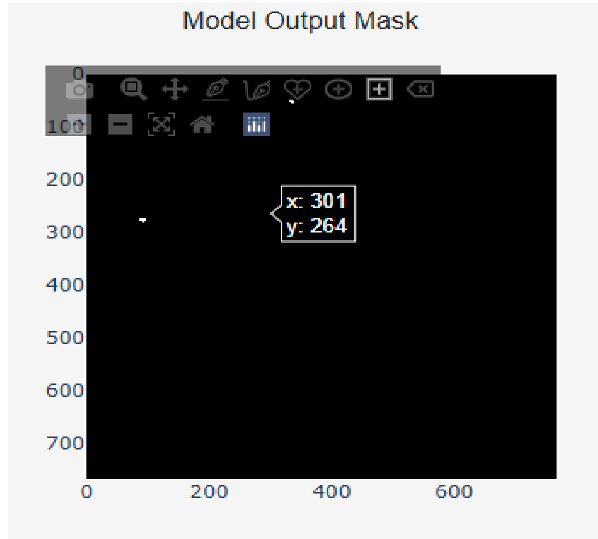












FUTURE ENHANCEMENT

Data Integration Using Multiple Sensors:

Investigate the use of data from many sensors, such as aerial radars, AIS, and marine radars. A comprehensive approach that incorporates data from multiple sources can improve the overall accuracy and reliability of ship monitoring systems.

Data Fusion and Analysis in Real-Time:

Use real-time data fusion techniques to merge data from several sensors. Advanced analytics and fusion algorithms can provide a more comprehensive and accurate picture of the maritime environment, allowing for faster decision-making.

Machine Learning to Detect Anomalies:

Use machine learning approaches, such as anomaly detection algorithms, to spot strange patterns or behaviors in ship movements. This can improve the system's ability to detect potential threats or abnormalities in near real time, hence contributing to better maritime security.

Continuous Training and Adaptation of Models:

Establish systems for ongoing model training and adaptability. Ship detectors can maintain their robustness and effectiveness in dynamic maritime situations by upgrading the Faster R-CNN models with new data and shifting circumstances.

Improving Object Recognition and Classification:

Invest in R&D to improve the object detection and classification capabilities of ship detectors. This could entail training the models on a larger dataset that includes a variety of vessel types and environmental circumstances.

Scalability of Larger Geographic Areas:

Create techniques for expanding the ship surveillance system across greater geographic areas. This can include deploying several sensors, developing algorithms for parallel processing, and assuring effective utilization of computer resources.

Incorporating Environmental Factors:

Consider using environmental elements such as weather and sea state in ship detection models. This added contextual information may increase the system's adaptability and reliability in a variety of maritime settings.

CONCLUSION

To summarize, this publication emphasizes the critical role of near real-time ship surveillance in ensuring maritime safety and security. Recognizing the limitations of traditional systems such as AIS and marine radars, particularly in scenarios where not all vessels have AIS transponders or visibility is limited, the use of airborne radars as supplementary sensors appears to be a promising solution, particularly in open sea environments. The development of unique object-oriented ship detectors that leverage the Faster Region-Based Convolutional Neural Network (R-CNN) demonstrates an innovative approach.

These detectors, which have been trained on large datasets and carefully tested in a variety of settings, operate well even in complex multitarget circumstances in the North Sea. However, future enhancements should take into account the integration of multi-sensor data, real-time data fusion, machine learning for anomaly detection, continuous model training, scalability for larger geographic areas, environmental factors, user-friendly interfaces with decision support, collaboration with maritime authorities, and robust cybersecurity measures. These improvements have the potential to improve ship surveillance capabilities, providing a complete and adaptable solution to meet the changing demands of maritime security in dynamic circumstances.

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 range compressed 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.