International Journal of Engineering and Advanced Technology Studies 12 (4), 35-47, 2024
Print ISSN: 2053-5783(Print)
Online ISSN: 2053-5791(online)
Website: https://www.eajournals.org/
Publication of the European Centre for Research Training and Development -UK

Deep Learning-Based Sonar Image Object Detection System

Md Shahazul Islam[1 *], A K M Jayed [2], Mohammad Rafiqul Islam[3], Almas Farhan Arnob[4], M A Alimul Hassan[5]
[1] Information and Communication Engineering, Nanjing University of Information Science and Technology, Nanjing, China
[2 ,4]Computer Science and Technology, Nanjing University of Information Science and Technology, Nanjing, China
[3] Computer Science and Technology, Xi'an Shiyou University, Xi'an, China
[5] Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing, China
*** Correspondence Email: shahazul5410@gmail.com | 202352180013@nuist.edu.cn**

doi: https://doi.org/10.37745/ijeats.13/vol12n43547 Published November 24, 2024

Citation: Islam M.S., Jayed A.K.M., Islam M.R., Arnob A.F., Hassan M.A.A. (2024) Deep Learning-Based Sonar Image Object Detection System, *International Journal of Engineering and Advanced Technology Studies* 12 (4), 35-47

Abstract: *Sonar image object detection is an important part of underwater exploration, submarine rescue, hostile object reconnaissance, and other critical maritime tasks. Accurate and efficient detection of objects in sonar imagery plays a key role in ensuring operational success in these domains. The breakthrough of deep learning-related technologies has brought new opportunities for the development of sonar image object detection. By leveraging advanced machine learning techniques, researchers have developed systems capable of achieving higher accuracy and robustness compared to traditional detection methods. However, despite these advancements, the relevant systematic research and practical applications remain insufficiently explored. Traditional approaches often struggle with challenges such as noise, low resolution, and the dynamic underwater environment, which limit their effectiveness. In contrast, deep learning models, with their data-driven advantages, have demonstrated significant potential in overcoming these challenges by learning robust feature representations from large-scale datasets. To address these gaps, a sonar image object detection system is designed to meet the requirements of accuracy, speed, portability, extensibility, and deployment adaptability in real-world scenarios. The system architecture is modular, consisting of three interdependent subsystems: dataset generation, algorithm model training and testing, and model deployment. The dataset generation subsystem ensures high-quality annotated sonar data, which is critical for effective model training. The training and testing subsystem incorporates state-of-the-art deep learning algorithms to optimize detection performance. Finally, the deployment subsystem focuses on translating the trained models into practical applications, ensuring they meet operational requirements under diverse environmental conditions. The system has been applied to underwater suspicious object detection* *tasks, addressing a range of scenarios requiring precise identification and localization of targets. The experimental results demonstrate that the object detection system achieves reliable and accurate performance, providing good test data and exhibiting excellent application outcomes. This work contributes to advancing the field of sonar image object detection, paving the way for future innovations in underwater exploration and related disciplines.*

Key words: sonar image; object detection system; deep learning; underwater suspicious object; deployment

INTRODUCTION

Imaging sonar uses the transmission and reception of sound signals for imaging, with a long detection range. It is currently a commonly used equipment for underwater exploration, underwater rescue, hostile target reconnaissance, and other tasks. Autonomous Target Recognition (ATR) of sonar images, also known as Object Detection, requires locating the area in the image that is most likely to contain the target and determining its category**Error! Reference source not found.**. Traditional object detection algorithms usually first use filtering methods such as sliding windows to list all possible bounding rectangle boxes of the target, and then use manually designed features such as edges and textures for classification, which cannot achieve good performance in complex and changing underwater environments.

In recent years, deep learning has made breakthrough progress by constructing feature extraction modules in a hierarchical manner, and constructing deep neural networks (DNNs) that are connected layer by layer to automatically learn image features through data-driven methods, overcoming the limitations of single artificial feature patterns and weak discriminative ability. The object detection algorithm based on deep learning has achieved end-to-end joint optimization of localization and classification tasks. Whether it is processing visible light natural images [1], or dealing with forward-looking sonar [2], side scan sonar [3], and synthetic aperture sonar images [4], it can achieve better detection accuracy than traditional methods [5][6].

However, sonar image acquisition requires a large amount of resources and is often not publicly disclosed due to the involvement of sensitive information [7]. Therefore, systematic research and application of deep learning based sonar image target detection are still insufficient. Practical engineering applications not only require high accuracy of detection algorithms, but also have certain requirements for software system quality attributes, running speed, deployment environment, and other aspects. Therefore, this article utilizes the data-driven advantages of deep learning models to design a sonar image target detection system, which does not necessarily indicate good performance in both test data and practical applications.

Sonar Image Object Detection System Based on Deep Learning

System composition

The designed sonar image object detection system based on deep learning includes three subsystems: dataset generation, algorithm model training and testing, and model deployment and application, as shown in Figure 1. The input and output of each subsystem are interrelated and do not have strong coupling relationships, meeting software system quality attributes such as portability, scalability, and ease of use. The designed system has universality and does not depend on a specific application task.

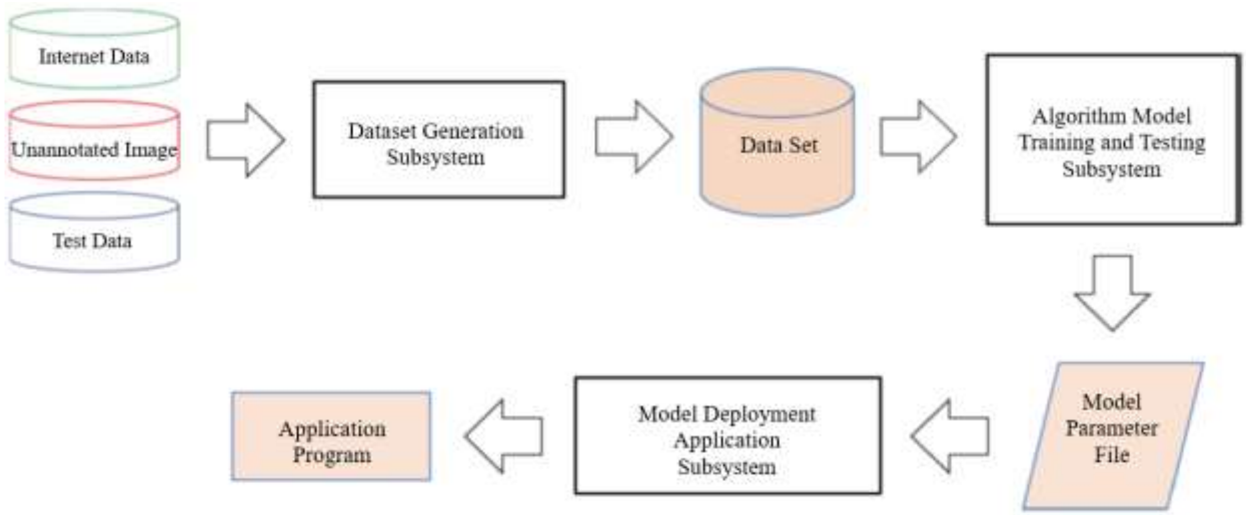

Fig. 1 Sonar image object detection system

The data set generation subsystem is responsible for collecting, labeling, processing, generating and managing data sets, labeling real-time data of field tests, existing unlabeled images and Internet data, generating data sets after preprocessing and training set test set division, and supporting modification, consolidation and other functions.

The algorithm model training and testing subsystem first constructs an object detection deep learning model, then reads the training set data for training, and reads the test set data to test the trained model. The model parameter file that meets the requirements of algorithm accuracy and speed is output.The model deployment application subsystem converts model parameter files according to the software and hardware environment requirements of different deployment platforms, and writes application programs that serve actual tasks. Then, it is deployed on the project machine, reads the model, and performs forward calculations.

Dataset generation subsystem

The data collection and annotation functions are divided into two modes: offline and online. Offline mode refers to the manual annotation of existing unlabeled sonar images. The data sources are usually historical data saved in field tests and open source images downloaded from the Internet. After collecting data, open source annotation software such as labellmg, labelme, Vott, CVAT, etc. can be used to annotate sonar images. Online mode refers to the real-time analysis of raw data transmitted by sonar equipment using relevant display software at the experimental site, and manually annotating and exporting the results after generating sonar images. Due to the known placement location of underwater targets during on-site testing, online annotation of category information is usually more accurate than offline annotation.

During the annotation process, it is advisable to avoid using narrow, extremely long to wide target bounding boxes for annotation, and to keep the target in the center of the bounding box to reduce irrelevant background information. The annotation format should also be consistent. As shown in Figure 2, P_1 is the top left vertex of the rectangular box, P_2 is the bottom right vertex, P_0 is the center point, and w, h represent the width and height of the rectangular box. Usually, annotations are recorded in data formats of $[x_1, y_1, w, h]$ or $[x_1, y_1, x_2, y_2]$. The YOLO series algorithm [8] normalizes the coordinates using the width and height of the image as denominators, and records the normalized center point coordinates and rectangle width and height.

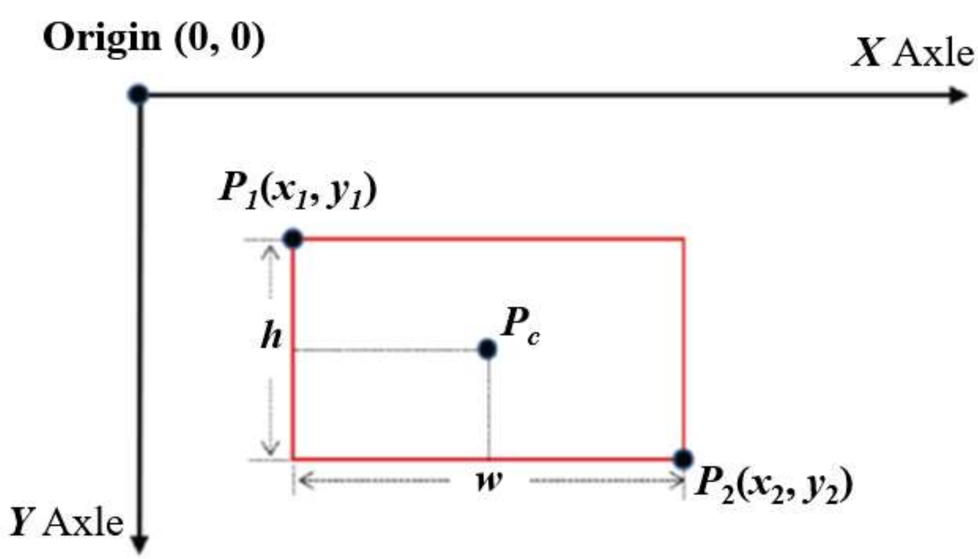

Fig. 2 An example of annotation coordinate

Traditional detection methods typically use filtering algorithms to preprocess sonar images, eliminate speckle noise, and improve detection accuracy [6]. Deep learning trains neural networks with data-driven approaches, and using noisy data to train models can improve algorithm robustness and resistance to attacks [9]. Therefore, this system does not perform filtering preprocessing operations on the dataset images.

Due to the difficulty in obtaining sonar images, the number of training images is usually small, which can easily lead to overfitting and lack of generalization in deep learning models. Therefore, data augmentation methods are crucial. The basic methods include geometric transformation operations such as flipping, rotating, cropping, deforming, scaling, etc., without modifying the content of the image itself. They are suitable for the sonar image dataset generation stage to increase the number of images.

Further enhancement methods typically use color space transformations to modify the semantic information of the image [10]. The impact of such methods on training results cannot be predicted in advance, so they are often attempted during the model training phase.Different types and models of sonar equipment and sonar image datasets collected in different water bodies should be classified and summarized. The commonly used sonar equipment includes forward-looking sonar, side scan sonar, and synthetic aperture sonar. Different types of sonar images have different styles. Forward looking sonar scans the fan-shaped area ahead with low image resolution and sensitivity to noise, while side scanning sonar and synthetic aperture sonar have high image resolution but lower image accuracy [5]. In addition, the distribution of sonar image data is also affected by factors such as water quality and underwater environment. The learnable information contained in data with different distributions varies, and the distribution of data in the source domain and target domain can affect the testing accuracy of deep learning models [11]. Therefore, the classification of sonar image datasets needs to consider various factors that affect data distribution.

Algorithm Model Training and Testing Subsystem

The algorithm model training and testing subsystem is typically deployed on servers configured with NVIDIA GPUs. The currently popular deep learning frameworks include PyTorch [12], TensorFlow [13], PaddlePaddle, etc. The widely used open-source object detection algorithm frameworks MMDetection [14], Detectron2, and YOLO series models [10] are all implemented based on PyTorch and Linux. Therefore, based on the above framework, this system defines the key module for constructing and training a deep learning model for sonar image object detection: Backbone. The backbone network is the main component of DNN for extracting image features. There are currently many high-performance Convolutional Neural Networks (CNNs), such as ResNet [15], DenseNet [16], ResNext [17], and Res2Net. Generally speaking, CNNs with stronger feature extraction capabilities have a larger number of parameters, resulting in slower inference speed. Therefore, lightweight CNNs are often used for high real-time tasks.

To further enhance feature richness, Neck networks typically use networks represented by Feature Pyramid Networks (FPNs) to construct multi-scale features[18], in order to improve the accuracy of small object detection in the model. In side scan sonar and synthetic aperture sonar images, the target to be detected often only occupies a small part of the entire image, so designing a neck network is crucial. The head network samples the extracted feature maps and calculates classification and localization results. The commonly used sampling methods currently include two-stage, one-stage, and anchor free. One stage of the method directly considers each coordinate point as a potential target, without additional candidate box extraction steps, and runs faster. Data augmentation is a data augmentation method based on color space transformation, commonly used in the training process, such as CutOut, CutMix, MixUp. These algorithms enrich the information of positive sample targets, alleviate the problem of insufficient sonar image data, and improve training efficiency and testing accuracy.

After the construction of the sonar target detection model is completed, the system reads the training set from the sonar image dataset for training, and then uses the test set for performance evaluation. For specific tasks, multiple different sonar image training sets can be selected for combination and comprehensively evaluated on multiple test sets. The model parameter files that meet the requirements of algorithm accuracy and speed can be output with a file extension of ". pt".

Similar to the evaluation indicators of MSCOCO [19] benchmark, the Intersection over Union (IoU) of the predicted box and the actual box is used to reflect the quality of a single prediction result**Error! Reference source not found.**. The result with IoU value and classification confidence greater than the established threshold is called True Positive (TP), which means it is correct; Otherwise, it is a False Positive (FP). Since the algorithm does not output negative sample background boxes, there is no True Negative (TN). False negative (FN) represents undetected target boxes, i.e. missed detections. Calculate the precision (P), recall (R), and average precision (AP) based on this:

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{1}$$

$$\mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{2}$$

$$\mathrm{AP} = \int_0^1 \mathrm{PdR} \tag{3}$$

The higher the IoU threshold and classification confidence set, the more stringent the criteria for predicting correctly. Usually, AP is calculated at intervals of 0.05 within the IoU threshold range of [0.5, 0.95], and the average score is taken as the comprehensive score. If the sonar image resolution is high but the target to be detected is small, the IoU interval can be set to [0.2, 0.65].

Model Deployment Application Subsystem

The model deployment application subsystem is responsible for deploying the trained sonar image object detection model to run in a specific environment. Due to issues such as dynamic modeling, implementation of new operators, and framework compatibility, model parameter files cannot be directly called when the runtime environment is different from the model training environment. The MMDeploy open-source deployment tool can be used to convert the model parameter files output by the algorithm model training and testing subsystems. MMDeploy supports running multiple algorithm libraries and converting model files in various formats, and supports Python, C++ interfaces and Windows, Linux operating systems.

After completing the conversion of the model parameter file, write an application to read and run it. The program includes the following functions:
1) The data receiving function obtains sonar image data to be processed from upstream programs through file reading, network transmission protocols, memory sharing, and other methods.
2) Target detection function: Read the converted model parameter file, load the memory/video memory, input the sonar image for calculation, and obtain the target detection result. According to the actual application effect, image preprocessing and result post-processing algorithms can be added.
3) Result sending function: Output detection results through file storage, network transmission protocol, memory sharing, and other methods.
The application program of the object detection system runs independently and interacts with upstream and downstream programs through established interfaces, meeting the principle of decoupling.

Implementation of Underwater Suspicious Target Detection System

This article applies the designed sonar image target detection system to underwater suspicious target detection tasks to verify its effectiveness.

The system uses a synthetic aperture sonar device from a certain company to conduct lake experiments in a fixed water area. Firstly, arrange multiple suspicious target shell models; Then, unmanned boat towing equipment is used to carry sonar equipment for data collection and annotation. During the navigation process, remotely control the unmanned boat to enter the mine laying area from different directions. Since the mine laying location is known in advance, controlling the direction of the unmanned boat's navigation ensures that any suspicious targets placed can appear within the sonar scanning range. The sonar image is rendered and synthesized by synthetic aperture sonar equipment and transmitted through line signals. The resolution of the collected original left sonar and right sonar images is 1900×1900. The positive sample annotation

only has one category of suspicious targets, and the position annotation is recorded based on the normalized center point coordinates and rectangle width and height.
Using geometric transformations such as scaling, cropping, stitching, and flipping, the original sonar images were processed to generate datasets with resolutions of 640×640, 1024×1024, and 3800×1900 (left and right sonar image stitching), totaling 1362 images. The specific division of the generated dataset and the number of images are shown in Table 1.

Table 1 Synthetic aperture sonar image dataset

Resolution	Number of Training Set Images/Sheet	Number of Test Set Images/Sheet
640×640	446	41
1024×1024	393	47
3800×1900	388	47

Considering the real-time requirements of system operation, the object detection model uses YOLOv5s network. The training images are enhanced using CutOut, MixUP, and Mosaic methods. The new image modified by combination is uniformly scaled to a resolution of 640×640 through adaptive scaling, with black borders added without changing the aspect ratio of the original information. The input image is processed through a backbone network to obtain multi-scale features with resolutions of 80×80, 40×40, and 20×20, and then expanded, overlaid, and enhanced in a neck network based on FPN. The output channels are 128, 256, and 512 layers of features, respectively. Then, the multi-scale features are input into the head network, and each pixel obtains a prediction result with 18 channels (4-dimensional represents the target coordinates, 1-dimensional represents the target confidence, 1-dimensional represents the suspicious target category result, with 3 preset anchor sizes, so 6×3=18). During the training process, both classification loss and confidence loss are calculated using the BCEWithLogitsLoss function, while coordinate regression loss is calculated using the GIOU method. The loss generated by multi-scale feature maps is weighted and fused to improve the accuracy of small object detection. The training parameters are all set to the default values of the YOLOv5 model. During the testing phase, weighted non maximum suppression is used to eliminate redundant prediction boxes, with an IoU threshold of 0.6 during the screening process.

After training, output the model parameter files that meet the accuracy requirements of the algorithm in ". pt" format. Due to the software and hardware environment constraints of the underwater suspicious target detection project, the system application program needs to be written in C++, and the deployed project computer is configured with Windows operating system, Inter

(R) Core (TM) i7-7700 CPU @ 3.60 GHz, 16 GB of memory, and NVIDIA GeForce RTX 2080 graphics card. PyTorch provides LibTorch Windows, a C++ based inference backend engine. Therefore, the system uses a model parameter conversion tool to convert the ". pt" format file to the ". torchscript. pt" format available for the engine.

The system application calls LibTorch related library functions to complete functions such as reading, loading, and inference of the converted model file parameters. Then, based on network transmission protocols, sonar data parsing, and other link libraries, develop sonar data receiving functions, as well as detection result sending and saving functions. At this point, the underwater suspicious target detection system has been implemented.

Experimental results

This chapter evaluates the detection performance of the system application using 135 synthetic aperture sonar images included in the test set. Tables 2 and 3 show the test results at IoU thresholds of 0.5 and 0.2, respectively.

Table 2 Experimental results of IoU=0.5

Classification Confidence Threshold	P/%	R/%	AP/%	AP@ 0.5 ~ 0.95/%
0.4	76.19	83.72	77.71	32.71
0.3	70.14	86.05	79.41	33.23
0.2	64.10	87.21	80.18	33.45
0.1	55.00	89.53	81.60	33.77
0.005	42.16	90.70	82.09	33.92

Table 3 Experimental results of IoU=0.2

Classification Confidence Threshold	P/%	R/%	AP/%	AP@ 0.2 ~ 0.65/%
0.4	81.45	89.53	85.75	77.38
0.3	74.88	91.86	87.56	78.77
0.2	68.38	93.02	88.38	79.42
0.1	58.21	94.77	89.49	80.34
0.005	44.86	96.51	90.26	80.88

According to the data in the table, when IoU is set to 0.2, the detection accuracy will significantly improve, indicating that there are a certain number of predicted boxes that deviate from their actual positions. The suspicious underwater targets in synthetic aperture sonar images do not overlap or appear densely, so the predicted results with a certain deviation have not been located incorrectly. This also indicates that there is room for further improvement in the localization performance of object detection algorithms.

When the set classification confidence is high, the accuracy P is high, and at this time, many low confidence prediction results are eliminated, reducing the false alarm rate (false alarm rate); However, due to the exclusion of low confidence results, the number of predicted boxes decreased, resulting in a decrease in the recall rate R and an increase in the missed detection rate (false alarm rate). When setting low classification confidence, although the average accuracy AP and comprehensive score will improve, the accuracy P will significantly decrease, resulting in a high false alarm rate. On the natural optical image dataset MS COCO, the classification confidence is usually set to 0.005 to achieve higher average accuracy. In this task, although setting the classification confidence level to 0.005 yields the highest average accuracy value of 82.09% (90.26%), it can lead to a high false alarm rate, causing frequent tension and fatigue among commanders in practical applications. In order to balance the false alarm rate and missed alarm rate, this system sets the classification confidence threshold to 0.3.

Figure 3 shows the application effect of the designed sonar image target detection system by selecting four images in the test set. The visualization results indicate that the system has high detection accuracy for suspicious underwater targets, but may generate false alarms for similar objects (as shown in the third image).

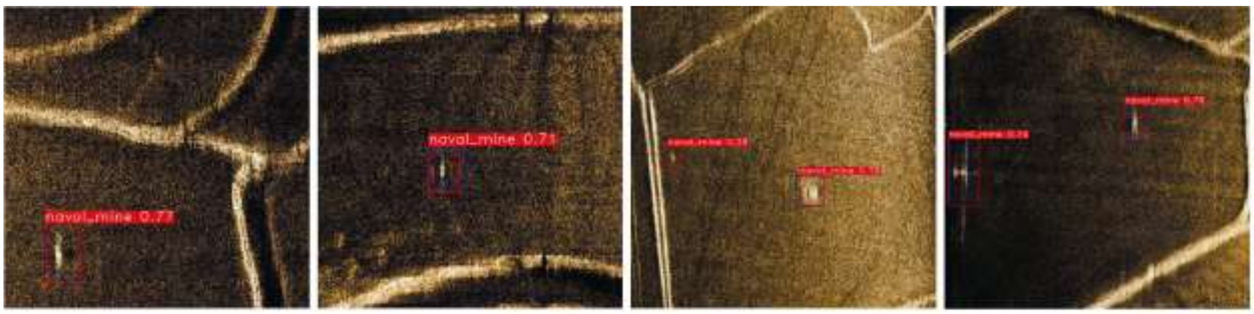

Fig.3 Detection effect of synthetic aperture sonar image test set

The initial model parameter size of the algorithm is 13.7 MB, which is converted to 27.3 MB. The system takes an average of 18 ms to process an image on an NVIDIA GeForce RTX 2080 graphics card, meeting real-time requirements.

CONCLUSION

This article proposes a sonar image target detection system based on deep learning, which demonstrates high universality and satisfies critical software system quality attributes, such as portability, scalability, and ease of use. By leveraging the advanced capabilities of deep learning, the proposed system addresses many of the limitations of traditional object detection methods in sonar imagery, making it suitable for a wide range of practical applications. The system has been effectively applied to underwater suspicious target detection tasks, showcasing reliable detection performance and achieving favorable results on synthetic aperture sonar images. Experimental findings highlight that fine-tuning the classification confidence threshold is critical to balancing detection accuracy and minimizing false alarm rates. Setting the threshold too high may result in missed detections, while excessively low thresholds can increase false positives, underscoring the importance of parameter optimization for real-world deployment. Beyond its current applications, the proposed system holds significant potential for broader use in both civil and military missions, including underwater infrastructure inspection, maritime archaeology, search and rescue operations, and national defense. Future enhancements to the system could focus on integrating adaptive algorithms to handle dynamic environmental conditions, incorporating real-time processing capabilities, and expanding its utility to diverse sonar imaging modalities. This work contributes to advancing sonar-based object detection technology and lays a strong foundation for further research and development in this field, with the ultimate goal of enabling safer and more efficient underwater operations across various domains.

References

[1] Lou, G., Zheng, R., Liu, M. and Zhang, S., 2020, October. Automatic target recognition in forward-looking sonar images using transfer learning. In *Global Oceans 2020: Singapore–US Gulf Coast* (pp. 1-6). IEEE.

[2] Lin, T., 2017. Focal Loss for Dense Object Detection. *arXiv preprint arXiv:1708.02002*.

[3] Fan, Z., Xia, W., Liu, X. and Li, H., 2021. Detection and segmentation of underwater objects from forward-looking sonar based on a modified Mask RCNN. *Signal, Image and Video Processing*, *15*(6), pp.1135-1143.

[4] Wang, Y., Liu, J., Yu, S., Wang, K., Han, Z. and Tang, Y., 2021, October. Underwater Object Detection based on YOLO-v3 network. In *2021 IEEE international conference on unmanned systems (ICUS)* (pp. 571-575). IEEE.

[5] Callow, H.J., 2003. Signal processing for synthetic aperture sonar image enhancement.

[6] Dong, B. and Wang, X., 2016, June. Comparison deep learning method to traditional methods using for network intrusion detection. In *2016 8th IEEE international conference on communication software and networks (ICCSN)* (pp. 581-585). IEEE.

[7] Kotwal, J., Kashyap, R. and Pathan, S., 2023. Agricultural plant diseases identification: From traditional approach to deep learning. *Materials Today: Proceedings*, *80*, pp.344-356.

[8] Hożyń, S., 2021. A review of underwater mine detection and classification in sonar imagery. *Electronics*, *10*(23), p.2943.

[9] Redmon, J., 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

[10] Zhang, H. and Wang, J., 2019. Towards adversarially robust object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 421-430).

[11] Bochkovskiy, A., Wang, C.Y. and Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

[12] Wang, M. and Deng, W., 2018. Deep visual domain adaptation: A survey. *Neurocomputing*, *312*, pp.135-153.

[13] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L. and Desmaison, A., 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, *32*.

[14] Dean, J. and Monga'TensorFlow, R., 2015. Large-Scale Machine Learning on Heterogeneous Distributed Systems'. *TensorFlow. org*.

[15] Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J. and Zhang, Z., 2019. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*.

[16] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[17] Xie, S., Girshick, R., Dollár, P., Tu, Z. and He, K., 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1492-1500).

[18] Gao, S.H., Cheng, M.M., Zhao, K., Zhang, X.Y., Yang, M.H. and Torr, P., 2019. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, *43*(2), pp.652-662.

[19] Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117-2125).

[20] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C.L., 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13* (pp. 740-755). Springer International Publishing.

[21] Islam M.S., Haque M.M., Salman M., Ali M.S., and Purokayisto S. (2024) Research on the Identification Algorithm of Crayfish Body Features Based on the Improved Yolov8n Loss Function, *International Journal of Engineering and Advanced Technology Studies*, 12 (4), 17-34