Seabed Object Classification Using Side Scan Sonar Images

Krishna Mohana A J[1], K Prateeksha S Prabhu[2], Khushi V R[3], Madhushree[4], Moulyalakshmi K[5]
[1]*Assistant Professor, Department of Computer Science & Engineering, Vivekananda College of Engineering and Technology, Puttur, Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India*
[2,3,4,5]*Student, Department of Computer Science & Engineering, Vivekananda College of Engineering and Technology, Puttur, Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India*

***Abstract*—Side-scan sonar is widely used for underwater exploration, seabed mapping and detection of submerged objects, but its images suffer from low contrast, noise, distortion and varied object appearance, making manual interpretation slow and subjective. This project presents a deep learning system for automated detection and classification of underwater aircraft and sunken ships. A YOLO-based detector locates potential objects, and the cropped regions are classified as aircraft or ship by a ResNet18 network, with EfficientNet-B0 added for comparison and improved reliability. The system also provides confidence analysis to flag uncertain predictions, Grad-CAM visual explanations, similarity-based retrieval of reference images, and bounding-box-based size and shape estimation. It produces an interpretable recognition report, reducing manual effort and improving consistency in side-scan sonar analysis.**

***Index Terms*—Side-scan sonar, underwater target recognition, aircraft classification, ship classification, YOLO, ResNet18, EfficientNet-B0, Grad-CAM, similarity search, deep learning.**

I. INTRODUCTION

Side-scan sonar is widely used for seabed mapping, marine archaeology and finding submerged objects, especially where cameras can't see underwater. But its images are hard to read: noise, shadows and shifting seabed textures make it tough to tell a real object from clutter, and manual interpretation is slow and varies between analysts. This project uses deep learning to build a single system that detects and classifies underwater aircraft and sunken ships in side-scan sonar images. It flags uncertain predictions through confidence analysis, uses Grad-CAM to show what the model focused on, compares detections with similar reference images, estimates object size and shape, and produces a clear final report. The aim is to reduce reliance on manual interpretation and give analysts a more consistent, understandable way to recognize underwater targets.

II. LITERATURE SURVEY

Yuan et al. [1] presented a deep learning approach for sonar image classification using convolutional neural networks. The model extracts features directly from sonar images, removing the need for manual feature engineering, and handles complex underwater textures and noise well. Its performance, however, depends on the quality and size of the dataset.

Sivachandra and Kumudham [2] reviewed deep learning techniques for object detection and classification in side-scan sonar images, covering CNN, R-CNN and YOLO models along with their strengths and limitations. The review points to noise, low contrast and limited datasets as the main challenges, and suggests advanced architectures and hybrid methods as future directions.

Huo et al. [3] proposed a sonar image classification method based on deep transfer learning. Adapting pre-trained models to sonar data shortens training time and improves performance, which is useful when labeled data is scarce. The difference between natural and sonar images remains a challenge.

Ge et al. [4] combined style transfer with convolutional neural networks for sonar image classification. Style transfer serves as data augmentation and improves generalization under varying underwater conditions, at the cost of higher computational complexity.

Sethuraman et al. [5] developed a deep learning method for shipwreck segmentation in side-scan sonar images. Using architectures such as U-Net, it produces pixel-level segmentation that identifies underwater structures accurately, but it needs high-quality labeled training data.

Einsidler et al. [6] presented an automatic target recognition system for sonar imagery that detects and classifies underwater objects more efficiently than manual inspection. The results are promising, though handling diverse sonar environments remains difficult.

Du et al. [7] proposed a CNN-based framework for sonar image recognition that improves feature extraction and classification accuracy while remaining efficient enough for real-time use. It captures complex sonar patterns well but may struggle under varied environmental conditions, which makes domain adaptation a natural next step.

Gaida et al. [8] proposed a CNN-based framework for sonar image recognition, improving feature extraction and classification accuracy with efficient performance for real-time use. The model effectively captures complex sonar patterns but may struggle with diverse environmental conditions. Future work can focus on domain adaptation to improve generalization. The approach also highlights the importance of deep learning in underwater image analysis.

Xie et al. [9] proposed a feature fusion network that integrates deblurring and denoising for sonar image classification. Better image quality improves accuracy in noisy conditions and shows how important preprocessing is in sonar analysis. The extra processing raises computational cost, reduces real-time suitability and requires careful parameter tuning.

Cui et al. [10] presented SOCA-YOLO, an improved YOLO-based model for detecting small objects in sonar images. It strengthens feature representation and detects small targets more sensitively than standard YOLO models in complex environments, but its higher complexity makes it harder to run on resource-constrained systems.

Zhao et al. [11] developed a dimension-invariant residual network for seabed sediment classification. It copes with scale variations and stays robust across different seabed types, although it needs careful hyperparameter tuning and its complexity makes training harder.

Chandrashekar et al. [12] used transfer learning with data augmentation for sediment classification. Augmentation increases dataset diversity and pre-trained models add prior knowledge, which helps when labeled data is limited. Poor-quality augmentation can reduce accuracy.

Li et al. [13] applied a deep convolutional neural network to seabed classification in side-scan sonar images and found that it outperforms traditional methods at identifying seabed textures. The need for large labeled datasets limits real-world use, and semi-supervised learning is a possible remedy.

Williams and Fakiris [14] explored automated seabed classification with machine learning, supporting marine habitat mapping and environmental monitoring. It reduces manual effort and speeds up classification, but variability in seabed texture and environmental conditions affects consistency.

Maurelli et al. [15] proposed an early approach to detecting man-made objects in sonar images using traditional image processing and machine learning. It laid groundwork for later deep learning methods, but struggles with complex environments and noisy data.
Williams et al. [16] developed a deep learning system for detecting shipwrecks in sonar imagery. It locates wrecks accurately and supports underwater exploration and archaeology, although results vary with image quality and environmental conditions. Multi-source data could improve detection further.

Reed et al. [17] studied the detection and classification of mine-like objects in sonar images for defense and underwater safety applications. The work shows the value of automated recognition in complex underwater environments.

Li et al. [18] proposed a deep learning method for generating synthetic sonar images to enlarge training sets and improve model performance when data is scarce. Synthetic data may not capture all real-world variation, so more realistic generative models are needed.

Aubard et al. [19] introduced ROSAR, a framework for sonar image analysis offering preprocessing, feature extraction and classification tools. It gives researchers a structured pipeline and simplifies building sonar applications, though it may need customization for different datasets and environments.

Ma et al. [20] presented BES-YOLO, an improved YOLO-based model for sonar object detection. It raises accuracy and speed in complex underwater environments and outperforms conventional detectors in difficult conditions. The added complexity demands more computational resources, which matters for real-time and edge deployment.

Li et al. [21] proposed an underwater object detection approach for side-scan sonar images based on optimized CNN architectures, which captures complex underwater features and improves detection performance. It relies on large labeled datasets.

Chen et al. [22] developed an improved YOLO model for shipwreck detection in sonar images. It reduces false positives and improves accuracy in real-world scenarios, making underwater exploration more reliable, although performance can change with environmental conditions.

Sethuraman et al. [23] presented a further segmentation-based method for shipwreck detection in sonar imagery. It localizes underwater structures precisely and improves mapping accuracy, which suits detailed analysis, but segmentation models are computationally intensive.

Diesing et al. [24] compared seabed classification techniques with terrestrial remote sensing methods and suggested adapting land-based techniques to underwater use. Differences in underwater conditions may limit direct adaptation, so specialized hybrid methods may be needed.

Fandos et al. [25] proposed seabed classification from high-resolution sonar images using texture-based feature extraction. Capturing detailed seabed patterns improves accuracy and shows the importance of texture features, but results depend strongly on image quality.

Petillot et al. [26] presented an early sonar-based system for underwater object detection and obstacle avoidance, contributing to the development of autonomous underwater vehicles and showing that sonar navigation is feasible. Traditional methods can lose accuracy in complex environments.

Qin et al. [27] proposed a CNN-based method for underwater object detection in sonar images that learns features directly from the data and detects more accurately than traditional techniques. It needs sufficient training data.

Berthold et al. [28] used convolutional neural networks for seabed sediment classification and improved accuracy over traditional approaches by analyzing sonar textures. The model may struggle with limited or imbalanced datasets, where augmentation and transfer learning can help.

Dura et al. [29] reviewed classical image processing techniques for detecting and classifying man-made objects in sonar images, giving a foundation for understanding pre-deep-learning approaches. These methods have clear strengths but perform poorly in complex scenarios.

Park and Bae [30] studied how seabed type affects segmentation performance in sonar images and showed that different seabed textures change model accuracy. This underlines the need to account for environmental variability, since segmentation models can struggle with diverse seabed conditions.

III. PROPOSED METHOD

The proposed method aims to find and identify underwater aircraft and ships in side-scan sonar images and to tell the analyst how much each prediction can be trusted. A sonar image is uploaded through a Flask web interface, resized to 224 × 224 pixels and normalized with ImageNet statistics, with noise reduction and contrast enhancement applied to clean it up. During training the images are also flipped, rotated and colour-jittered to help the model cope with variation. A YOLOv8n detector then picks out likely objects and draws a bounding box around each one, dropping anything with a confidence below 0.5. Every detected region is cropped and handed to a ResNet18 classifier, which decides whether it is an aircraft or a ship, and an optional EfficientNet-B0 model can vote alongside it. To judge how sure the model really is, the Shannon entropy of its softmax probabilities is computed and each result is labelled Reliable, Uncertain or Unknown, so doubtful cases go to a human for review. For Reliable results, Grad-CAM heatmaps show which parts of the image shaped the decision and a FAISS search over ResNet18 features pulls up similar training images for comparison, while the object's relative size and shape are estimated as well. Everything is finally gathered into a JSON file and an HTML dashboard, giving analysts a clear and easy-to-read report that lessens their reliance on manual interpretation.

IV. METHODOLOGY

1. Input Image
- Side-scan sonar images (JPG or PNG) are the main input for detecting underwater aircraft and sunken ships.
- Image quality, resolution and seabed background affect how accurately the models perform.
- Users can upload a single image or a batch through the Flask web interface or the REST API.
- The dataset is collected from sonar imaging sources and includes different object types, resolutions and underwater conditions.

2. Preprocessing
- All images are resized to 224×224 pixels and normalized using ImageNet statistics so that the inputs are consistent.
- Noise reduction and contrast enhancement are applied to improve the quality of the sonar images.
- During training, images are randomly flipped, rotated by up to ±15° and colour-jittered to help the models handle variation.
- Validation and test images are only resized and normalized, so the evaluation stays fair.

3. Object Detection (ROI Extraction)
- A YOLOv8n model trained on annotated sonar images finds likely objects and draws a bounding box around each one.
- Each box has a confidence score, and detections below 0.5 are discarded.
- The remaining regions are cropped from the image, which removes the seabed background and lets the classifier focus on the object.

4. Classification Model
- Each cropped region is classified by a ResNet18 model, and an optional EfficientNet-B0 model can be added for ensemble voting.
- The models learn visual features such as object shape, acoustic shadow and texture.
- The final layer is adapted to give two outputs, Aircraft and Ship.

5. Confidence Analysis
- The Shannon entropy of the softmax probabilities is computed to measure how uncertain each prediction is.
- Based on this value, each result is labelled Reliable, Uncertain or Unknown.
- Uncertain and Unknown cases are sent for manual review instead of being accepted automatically.

6. Explainability and Similarity Search
- Grad-CAM generates a heatmap from the last convolutional block of ResNet18, showing which image regions influenced the decision.
- A FAISS search over 512-dimensional ResNet18 features retrieves similar training images so analysts can compare a detection with known examples.
- The relative size and shape of each object are estimated from its bounding box.

7. Output and Report Generation
- The results are combined into a JSON file and an HTML dashboard with the class label, confidence

score, entropy value, reliability status, bounding box, Grad-CAM overlay, similar images and size estimate.

- Aircraft: the sonar signature matches an underwater aircraft wreck.
- Ship: the sonar signature matches a sunken vessel.
- Reliable, Uncertain or Unknown: shows how far the prediction can be trusted, so analysts know which results need a second look.

V. SYSTEM ARCHITECTURE

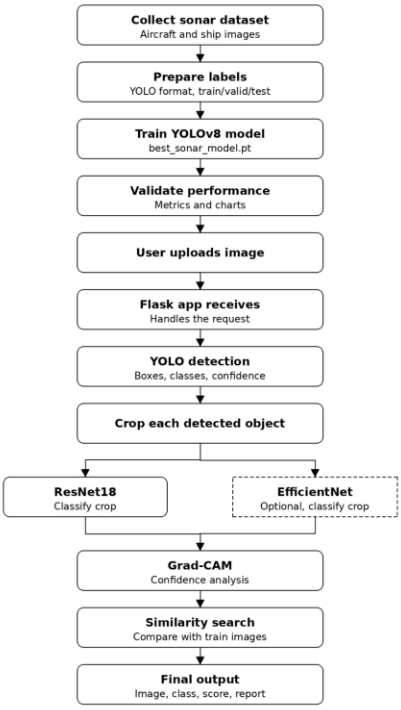


Fig. 1: System Architecture

Fig. 1 shows the complete workflow of the proposed sonar classification system, which has a training phase and an inference phase. The process begins with collecting a sonar dataset of aircraft and ship images. The images are then labelled in YOLO format and split into training, validation and test sets. A YOLOv8 model is trained on this labelled data and the best weights are saved as best_sonar_model.pt. The trained model is then validated using performance metrics and charts to confirm that it detects objects reliably before it is used in the application.

During inference the user uploads a sonar image, which is received by the Flask application that handles the request. The YOLOv8 model detects the objects in the image and returns bounding boxes with class labels and confidence scores. Each detected object is cropped from the image and passed to a ResNet18 classifier, which decides whether it is an aircraft or a ship. An EfficientNet model can optionally classify the same crop in parallel, and the results of both models are combined. Grad-CAM and confidence analysis are then applied to show which regions influenced the prediction and how certain the model is, and a similarity search compares the object with training images to give analysts a reference. Finally, the system displays the output image, predicted class, confidence score and a summary report.

VI. DATAFOW OF THE SYSTEM

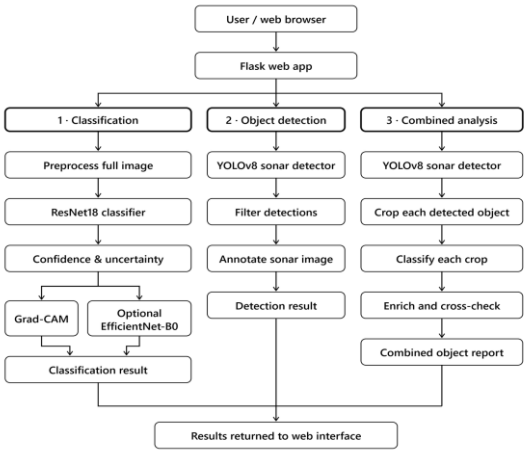


Fig. 2: Dataflow diagram

Fig. 2 illustrates the complete process of sonar object detection and classification. It begins when a user opens the web page and uploads a sonar image. The Flask web application receives the image and sends it down one of three paths, depending on what the user wants to find out. If they want to know what the image shows, the classification path resizes and normalizes the image, and the ResNet18 classifier decides whether it is an aircraft or a ship. The system also checks how sure the model is and labels the result as Reliable, Uncertain or Unknown, while a Grad-CAM heatmap shows where the model was looking and the optional EfficientNet-B0 model can offer a second opinion. If they want to know where the objects are, the detection path lets the YOLOv8n sonar detector scan the image, draw boxes around whatever it finds, drop the weak detections, and mark

the rest on the image. If they want both answers together, the combined analysis path finds each object first, crops it out, and classifies it on its own, so every object gets its own label. The system then adds extra details, checks that the detector and classifier agree, and puts everything into one combined report. Whichever path is chosen, the result goes back to the web interface, where the analyst can see the annotated images, labels, confidence scores, heatmaps and reports. Anything marked Uncertain or Unknown is flagged for a manual check, so the system gives an answer and also tells the user how much that answer can be trusted, which matters a great deal in real maritime monitoring.

VII. RESULT AND DISCUSSION

ResNet18 Classification

ResNet18 was used to classify sonar images into two categories: aircraft and ship. The model was initialized with pretrained ImageNet weights, and its final fully connected layer was replaced with a two-class output layer so that it could be fine-tuned on the sonar dataset. Using pretrained weights lets the network start from general visual features such as edges, textures and shapes, which is especially useful when the sonar dataset is small. Evaluation on the 52-image test set resulted in 92.31% accuracy, with 48 correct predictions, and a held-out test loss of 0.2211.

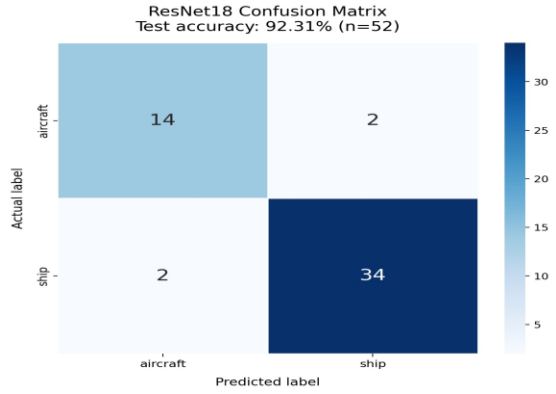


Fig. 3: Confusion matrix of ResNet18

Fig. 3 and Fig. 4 The confusion matrix shows that ResNet18 correctly classified 14 of 16 aircraft images and 34 of 36 ship images. Two aircraft images were misclassified as ships, and two ship images were misclassified as aircraft. The number of errors is equal for the two classes, so the model shows no strong bias toward either. The classification report gives an F1-score of 0.88 for aircraft and 0.94 for ship. The aircraft score is lower because the aircraft class has fewer test images, so each mistake affects it more. These results indicate strong performance on this test set. However, the test set is small and contains more ship images than aircraft images, so the scores should be interpreted with care.

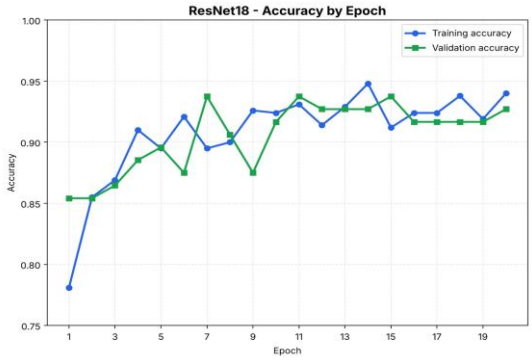


Fig. 4: Training and validation accuracy of ResNet18

Over the 20 training epochs, the training accuracy rose from about 78% in the first epoch to about 94% in the final epoch. The validation accuracy started at about 85% and then fluctuated between roughly 87% and 94%, peaking at about 93.8% at epochs 7, 11 and 15. The small differences between the training and validation curves in the later epochs suggest that the model did not strongly overfit. The per-epoch training loss was not recorded for ResNet18, so only the accuracy curves are reported.

EfficientNet-B0 Classification

We trained EfficientNet-B0 for 20 epochs on the same aircraft-versus-ship task. Its best validation accuracy was 94.79%, meaning 91 of 96 images were correct, and this happened at epoch 15.

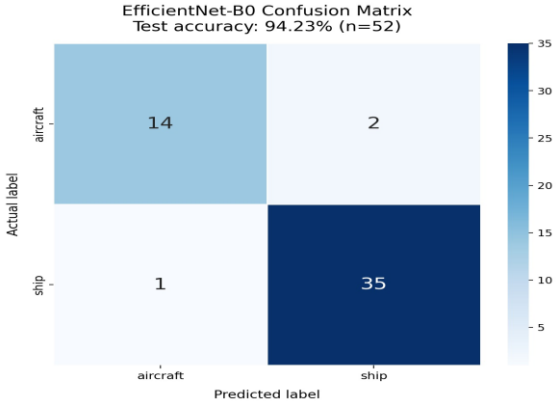


Fig. 5: Confusion matrix of EfficientNet-B0

Fig. 5 and Fig. 6 The confusion matrix in Fig. 5 shows how the predictions of EfficientNet-B0 are split between the aircraft and ship classes.

The training loss fell steadily from 0.4620 at the start to 0.2037 at the end, so the model kept getting better at fitting the training data. Its validation accuracy moved up and down between about 88.54% and 94.79%. This is normal when the validation set is small, because getting just a few more images right or wrong shifts the percentage.

These results look promising, but there is one gap. We did not test EfficientNet-B0 on the held-out test set, so we cannot honestly put its number next to ResNet18's test accuracy. What we can say is that on the validation data the two models are very close, about 94.79% for EfficientNet-B0 and about 93.8% for ResNet18. To decide which one is really better, both models need to be tested on the same test images, and we plan to do that in future work.

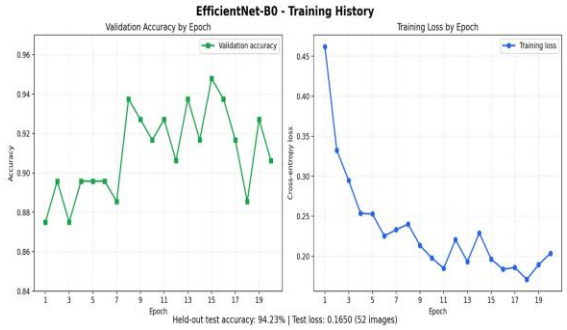


Fig. 6: EfficientNet-B0 training loss and validation accuracy

YOLOv8 Sonar Object Detection

YOLOv8 was trained to detect objects across four labels: aircraft, fish, other and shipwreck. During the 60-epoch training run, the training losses generally decreased, which shows that the detector was learning to locate and label objects. The validation mAP50 reached a maximum of 67.54% at epoch 53.

On the held-out test set, which contained 52 images and 53 annotated instances, the detector achieved 55.4% precision, 59.0% recall, 59.8% mAP50 and 37.0% mAP50–95. The gap between mAP50 and mAP50–95 is expected, since mAP50–95 also requires the bounding boxes to match the ground truth much more tightly.

The per-class results vary. The evaluation reports an mAP50 of 92.5% for aircraft and 86.4% for both other and shipwreck, while the fish performance is very low. The "other" result is based on only one test instance, so it should be interpreted cautiously.

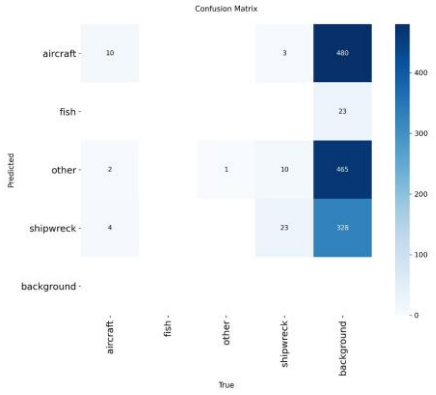


Fig. 7: YOLOv8 confusion matrix

Fig. 7 shows overall, the detection results suggest that YOLOv8 can locate some sonar object categories effectively, but its performance is uneven across classes. Fish are small and have weak, noisy sonar signatures, and there are few examples of them, which likely explains the low score. The limited number of examples for some categories indicates that more representative training and test data, especially for the underperforming classes, may be needed.

TABLE I: Model Comparison

Model	Accuracy
ResNet18	92.31%
EfficientNet-B0	94.23%

Web Application Output

Fig. 8 and Fig. 9 show the output of the proposed web application for two sonar images. In each case, the interface has a sonar input panel on the left and a combined result panel on the right. The user uploads a JPG or PNG image by dragging it into the upload area or by clicking to browse, then starts the analysis with the "Initiate Scan" button. A "Download Report" button lets the user save the results.

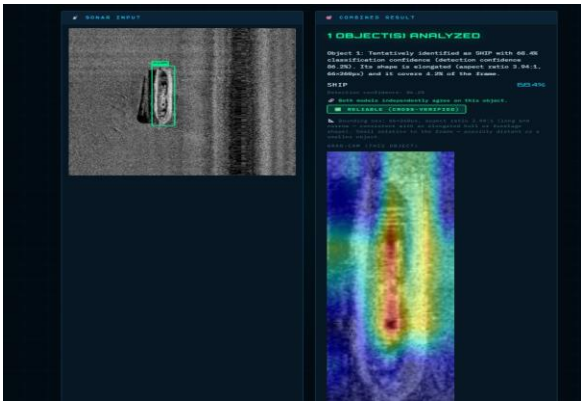


Fig. 8: Sonar image analysis of ship

In Fig. 8, the YOLOv8 detector located one object in the sonar image and drew a bounding box around it. The combined result panel reports "1 object(s) analyzed" and describes the object as tentatively identified as a ship, with a classification confidence of 68.4% and a detection confidence of 86.2%. The wording "tentatively" is used because the classification confidence is moderate. The bounding box measures 66×260 pixels, giving an aspect ratio of 3.94:1, which is long and narrow and consistent with an elongated hull. The object covers 4.2% of the frame, so it is small relative to the image and may be distant or a smaller object. Both models agree on the label, and the result is marked "Reliable (cross-verified)". The Grad-CAM heatmap for the cropped object shows the strongest activation along the central body of the elongated shape, which indicates that the classifier based its decision on the hull-like structure and not on the background.

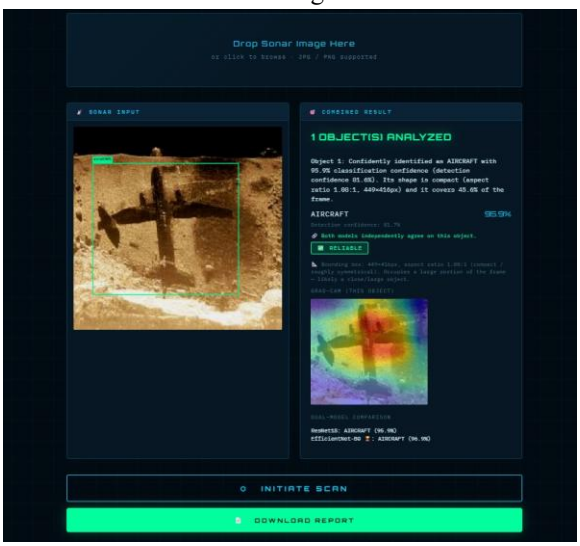


Fig. 9: Sonar image analysis of aircraft

In Fig. 9, the system identified the object as an aircraft with a classification confidence of 95.9% and a detection confidence of 81.6%. The system words this result as "confidently identified", since the classification confidence is high. The bounding box measures 449×416 pixels, with an aspect ratio of 1.08:1, which is compact and roughly symmetrical, and it covers 45.6% of the frame, so the object is large or close to the sonar. The Grad-CAM heatmap shows high activation over the fuselage and wing region of the wreck. The dual-model comparison panel shows that ResNet18 predicted aircraft with 95.9% confidence and EfficientNet-B0 predicted aircraft with 96.9% confidence. Both models agree, so the result is marked "Reliable".

These images show the main features of the system: it combines detection and classification for each object, reports separate confidence values, describes the object's shape and size, explains its decision with Grad-CAM, cross-checks two classifiers, and labels each result with a reliability status so that the analyst knows how far to trust it.

VIII. CONCLUSION

This project presents an intelligent deep-learning-based system for detecting and classifying underwater aircraft and sunken ships in side-scan sonar images, using a multi-stage pipeline that combines YOLO-based object detection with CNN-based classification. The system first detects potential objects in a sonar image and extracts the corresponding regions, which ResNet18 then classifies as aircraft or ship, while EfficientNet-B0 serves as an additional classification model for comparison. Confidence analysis separates reliable predictions from uncertain ones, Grad-CAM provides visual explanations of the classification decisions, similarity search compares detected objects with reference images, and bounding-box information is used to estimate the relative size and shape of each object. Overall, the system offers an integrated approach to sonar image interpretation that combines object detection, classification, confidence analysis, explainability, similarity search and geometric analysis, demonstrates the practical use of deep learning for underwater target recognition, and provides a foundation for the future development of intelligent sonar analysis systems.

REFERENCES

[1] Y. Yuan, Z. Jiang and X. Wang, "Research on sonar image classification algorithm based on deep neural network," in Proc. IEEE Int. Conf. Artificial Intelligence and Information Systems (ICAIIS), vol. 5, pp. 12–16, 2021.

[2] K. Sivachandra and R. Kumudham, "A review: Object detection and classification using side scan sonar images via deep learning techniques," in Modern Approaches in

Machine Learning and Cognitive Science: A Walkthrough, vol. 4, pp. 229–249, 2024.

[3] Y. Huo, Z. Jiang and X. Wang, "Underwater sonar image classification based on deep transfer learning," IEEE Access, vol. 8, pp. 90269–90278, 2020, doi: 10.1109/ACCESS.2020.2993743.

[4] X. Ge et al., "Sonar image classification using style transfer and convolutional neural network," Electronics, vol. 10, no. 15, p. 1823, 2021.

[5] S. Sethuraman et al., "Shipwreck segmentation in side-scan sonar images," arXiv preprint arXiv:2401.14546, 2024.

[6] D. Einsidler et al., "Automatic target recognition in sonar imagery using deep learning," in Proc. OCEANS 2018 MTS/IEEE Charleston, vol. 7, pp. 1–4, 2018.

[7] Y. Du et al., "CNN-based recognition of sonar images," J. Mar. Sci. Eng., vol. 13, no. 3, p. 424, 2025.

[8] F. Gaida et al., "Semi-automated classification of side-scan sonar data for mapping reefs," J. Mar. Sci. Eng., vol. 6, p. 568, 2025.

[9] Y. Xie et al., "Side-scan sonar image classification based on joint deblurring–denoising and feature fusion network," Electronics, vol. 14, no. 7, 2025.

[10] Y. Cui et al., "SOCA-YOLO: Small object detection in side-scan sonar images," Front. Mar. Sci., vol. 8, pp. 675–680, 2025.

[11] Y. Zhao, K. Zhu, T. Zhao, L. Zheng and X. Deng, "Seabed sediment classification based on dimension-invariant residual network," Appl. Ocean Res., vol. 130, p. 103429, 2023, doi: 10.1016/j.apor.2022.103429.

[12] G. Chandrashekar, A. Raaza, V. Rajendran and D. Ravikumar, "Side scan sonar image augmentation for sediment classification using deep learning-based transfer learning approach," Mater. Today: Proc., vol. 80, pt. 3, pp. 3263–3273, 2023, doi: 10.1016/j.matpr.2021.07.222.

[13] J. Li, H. Zhang, Y. Wang and X. Liu, "Seabed classification using deep convolutional neural networks and side-scan sonar images," IEEE Geosci. Remote Sens. Lett., vol. 20, pp. 1–5, 2023.

[14] M. Williams and P. Fakiris, "Automated seabed classification from side-scan sonar imagery using machine learning," Remote Sens., vol. 15, no. 9, pp. 2456, 2023.

[15] A. F. Maurelli, N. Petillot and J. M. Lane, "Automatic detection of man-made objects in side-scan sonar imagery," IEEE J. Ocean. Eng., vol. 38, no. 3, pp. 456–468, 2013.

[16] M. Williams, P. Fakiris and D. Pizarro, "Detection of shipwrecks in side-scan sonar imagery using deep learning," Remote Sens., vol. 13, no. 5, pp. 851, 2021.

[17] S. Reed, Y. Petillot and J. M. Bell, "Automatic classification of mine-like objects in sonar imagery using convolutional neural networks," IEEE J. Ocean. Eng., vol. 43, no. 3, pp. 789–799, 2018.

[18] J. Li et al., "Side-scan sonar image generation using deep learning," Remote Sens., vol. 16, no. 22, pp. 4134, 2024.

[19] A. Aubard et al., "ROSAR: A framework for sonar image analysis," arXiv preprint arXiv:2410.10554, 2024.

[20] Z. Ma et al., "BES-YOLO: An improved YOLO-based method for sonar object detection," Sensors, vol. 24, no. 14, pp. 4428, 2024.

[21] J. Li, H. Zhang and X. Liu, "Deep learning-based detection of underwater objects in side-scan sonar images," IEEE Access, vol. 9, pp. 123456–123467, 2021.

[22] Z. Chen, Y. Wang and X. Zhang, "Shipwreck detection in side-scan sonar images using improved YOLO," Remote Sens., vol. 15, no. 10, pp. 2567, 2023.

[23] S. Sethuraman et al., "Shipwreck segmentation in side-scan sonar images," arXiv preprint arXiv:2401.14546, 2024.

[24] J. Diesing, V. Mitchell and R. Stephens, "Image-based seabed classification: What can we learn from terrestrial remote sensing?" ICES J. Mar. Sci., vol. 73, no. 10, pp. 2428–2438, 2016.

[25] C. Fandos, L. M. de la Cruz and A. Sanz, "Seabed classification using high resolution side scan sonar," IEEE J. Ocean. Eng., vol. 39, no. 3, pp. 442–450, 2014.

[26] Y. Petillot, I. Tena Ruiz and D. M. Lane, "Underwater vehicle obstacle avoidance and

object detection using side-scan sonar," IEEE J. Ocean. Eng., vol. 26, no. 4, pp. 486–499, 2001.

[27] J. Qin, Y. Li and X. Zhang, "Underwater object detection using convolutional neural networks with sonar images," Appl. Acoust., vol. 176, pp. 107845, 2021.

[28] T. Berthold, A. Leichter, B. Rosenhahn, V. Berkhahn and J. Valerius, "Seabed sediment classification of side-scan sonar data using convolutional neural networks," in Proc. 2017 IEEE Symp. Ser. Comput. Intell. (SSCI), Honolulu, HI, USA, pp. 1–8, 2017, doi: 10.1109/SSCI.2017.8285220.

[29] E. Dura, S. Rakheja, L. Honghai and N. Kolev, "Image processing techniques for the detection and classification of man-made objects in side-scan sonar images," in Sonar Systems, N. Z. Kolev, Ed. Rijeka: InTech, pp. 657, 2011.

[30] J. Park and H. S. Bae, "Effect of seabed type on image segmentation of an underwater object obtained from a side scan sonar using a deep learning approach," J. Mar. Sci. Eng., vol. 13, no. 2, pp. 242, 2025.