Ocean Eye: Advanced Neural Segmentation for High-Resolution Ship Detection

Y. Bhavani[1], T. Harika[2], T. Mahesh Babu[3], T. Naresh[4]
[1,2,3,4] UG Students, Department of CSE,
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.

Dr. A. Srinivasa Rao[5], Professor, Department of CSE,
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh

[1]22bq1a05o2@vvit.net, [2]22bq1a05l2@vvit.net,
[3]22bq1a05m2@vvit.net, [4]22bq1a05l1@vvit.net , srao.akella@vvitu.ac.in

Abstract - Maritime safety and monitoring have become more important with the increase in global trade and ship traffic. This paper introduces **Ocean Eye**, a smart maritime surveillance system that can detect ships, track their movement, measure real-world distances, and identify possible collision risks. The system uses **YOLOv8** for detecting objects, **DeepSORT** for tracking ships, and socket communication for handling real-time distress signals. It also supports restricted zone monitoring and provides a user-friendly dashboard for visualization. The proposed system improves maritime safety by delivering real-time alerts and helping in better decision-making, similar to modern monitoring systems.

Plain Language Summary: This research introduces Ocean Eye, a smart system that helps monitor ships and improve safety in oceans. The system uses artificial intelligence to detect ships and track their movement in real time. It can calculate distances between ships and identify possible collision risks. It also monitors restricted areas and sends alerts when rules are violated or danger is detected. By providing real-time insights and warnings, the system helps improve maritime safety and supports better decision-making.

Keywords – Maritime Surveillance, YOLOv8, DeepSORT, Collision Detection, Computer Vision, Ship Tracking, Socket Networking.

INTRODUCTION

1.1 Background and Motivation

Oceans play an important role in transportation, trade, and fishing activities. As the number of ships continues to grow, ensuring safety has become more challenging. When multiple ships operate in the same region, there is a higher chance of accidents, collisions, or unauthorized entry into restricted areas. Existing methods such as radar systems and manual monitoring are not always efficient and depend heavily on human involvement.

Delays in identifying risks or handling emergency situations can lead to serious consequences. In addition, continuously monitoring large ocean regions is difficult using traditional approaches. These issues highlight the need for a more advanced and automated solution.

The main goal of this project is to design an intelligent system that can detect ships, monitor their movement, and generate alerts whenever a risk is identified. The proposed system, **Ocean Eye**, enhances maritime safety by providing real-time monitoring and supporting faster and more effective decision-making.

1.2 Artificial Intelligence in Legal Document Analysis

With the advancement of artificial intelligence, many modern systems are using AI to monitor ship activities in oceans. Computer vision techniques allow automatic detection of ships from images and videos, which reduces the need for continuous human supervision.

Models such as YOLO can identify ships quickly, while tracking algorithms like DeepSORT help in following their movement over time. This makes it easier to understand the direction and behavior of ships.

AI-based systems are especially useful in situations where many ships are present. They can process data quickly, generate alerts, and support better decision-making, which helps in improving overall maritime safety.

1.3 Limitations of Existing Approaches

Although AI-based systems can detect ships, they are not completely reliable. Their performance may decrease in conditions such as poor lighting, bad weather, or when ships are very small or far from the camera. In addition, these systems often do not provide clear reasoning for their results, which makes them harder to trust.

On the other hand, traditional methods like radar are still useful, but they depend on human involvement and cannot perform advanced tasks such as tracking ship movement or predicting risks. They are also less effective when monitoring large ocean areas continuously.

Due to these limitations, existing approaches are not sufficient for accurate and real-time maritime monitoring. There is a need for a more complete and intelligent system that can handle detection, analysis, and decision-making together.

1.4 Need for a Neuro-Symbolic Approach

To address the limitations of using only AI or only traditional methods, a combined approach can be adopted. AI models are effective in detecting ships and analyzing visual information, while traditional systems provide reliable monitoring and support better decision-making.

Fig. 1 *Conceptual overview of the combined system showing AI-based detection and rule–based monitoring*

By combining these approaches, the system is able to go beyond basic ship detection and perform more meaningful analysis. It can track ship movement, evaluate situations, and generate clear alerts when any risk is identified, such as possible collisions or entry into restricted zones. The integration of AI with rule-based monitoring allows the system to make more accurate and reliable decisions.

This combination improves overall safety by reducing errors and ensuring timely warnings. It also helps users understand why a particular alert is generated, making the system more transparent and trustworthy. As a result, the proposed approach provides a strong and dependable solution for modern maritime surveillance systems.

1.5 Proposed Neuro-Symbolic Framework

Motivated by the challenges discussed earlier, this work proposes an intelligent maritime surveillance system named **Ocean Eye**. The system uses an AI-based model, YOLOv8, to detect ships and other objects from images and video streams. Once detected, the objects are tracked using DeepSORT, which helps in analyzing their movement and behavior over time.

The detected and tracked data is further processed using a rule-based module. This module calculates distances between ships, identifies possible collision risks, and checks whether any vessel enters restricted zones. In addition, the system supports communication through socket networking, enabling clients to send distress signals and receive alerts instantly.

By combining AI-based detection with rule-based analysis, the system improves overall accuracy and reliability. The AI component handles detection and tracking efficiently, while the rule-based logic ensures proper evaluation of conditions such as collision risk and zone violations. This integration allows the system to generate clear and meaningful alerts, making it suitable for real-time maritime monitoring and decision-making.

2. RELATED WORK

Earlier approaches in maritime surveillance mainly relied on traditional systems such as radar and manual observation. These methods were effective for basic ship detection and provided essential information about vessel presence. However, they required continuous human monitoring and were not capable of handling large-scale data efficiently. In addition, these systems lacked advanced capabilities such as ship tracking, behavior analysis, and prediction, which limited their usefulness in modern maritime environments [1].

With the development of machine learning techniques, researchers began introducing automated methods for ship detection and analysis. These approaches helped in identifying ships from images and extracting important features. While they improved the level of automation, they depended heavily on manually designed features and struggled to perform well in challenging conditions such as poor visibility, cluttered scenes, and varying ship sizes [2].

To overcome these limitations, deep learning-based models have been widely adopted in recent years. Models such as YOLO have significantly improved real-time object detection by providing faster and more accurate results. In addition, tracking algorithms like DeepSORT have been used to assign unique IDs to ships and monitor their movement across multiple frames. These advancements have enhanced the ability to track multiple vessels and analyze their behavior in real time [3].

Despite these improvements, most existing systems are limited to detection and tracking tasks. They do not include higher-level functionalities such as collision risk prediction, restricted zone monitoring, or real-time communication between systems. The lack of integration between these components reduces the effectiveness of current solutions and limits their ability to provide complete maritime surveillance [4].

To address these challenges, the proposed system, **Ocean Eye**, introduces an integrated approach that combines AI-based detection, object tracking, rule-based analysis, and communication features into a single platform. This system aims to provide accurate monitoring, early detection of risks, and timely alerts, thereby improving overall safety and decision-making in maritime environments.

3. METHODOLOGY

The proposed methodology focuses on developing an intelligent maritime surveillance system by combining artificial intelligence with rule-based analysis. The system integrates ship detection, tracking, and monitoring techniques to enhance safety in ocean environments. The overall workflow of the system is illustrated in Fig. 2, which presents a clear overview of the complete process.

The major stages of the system include:

• Input video/image acquisition
• Preprocessing (resize, frame extraction, noise removal)
• Ship detection using YOLOv8
• Object tracking using DeepSORT
• Ship classification (type identification)
• Distance calculation (GSD method)
• Collision risk detection
• Restricted zone monitoring
• Distress signal handling (client-server communication)
• Alert generation (warnings & notifications)
• GUI dashboard display

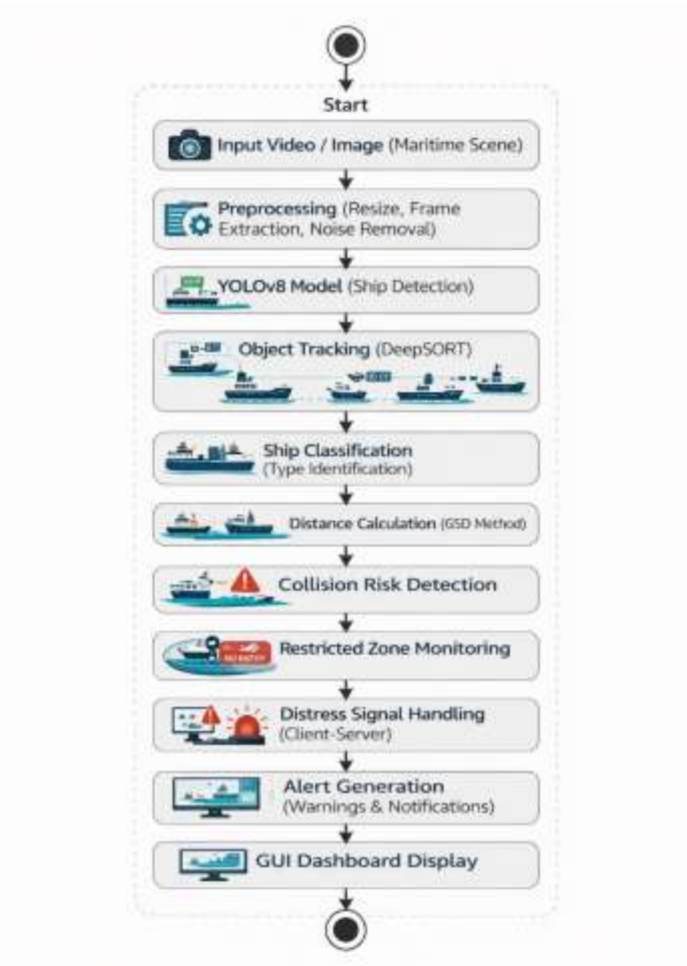

Fig. 2. Overall Workflow of the Proposed Ocean Eye System.

Fig. 2 illustrates the UML activity diagram of the proposed **Ocean Eye** maritime surveillance system. The process starts with input video or image data, followed by preprocessing, ship detection, object tracking, and classification. It then includes distance estimation, collision risk analysis, and monitoring of restricted zones, and finally produces alerts that are displayed on the GUI dashboard. This modular workflow supports efficient processing, enables real-time monitoring, and enhances safety in maritime environments.

A. Legal Document Dataset Acquisition and Preprocessing.

The quality of input data plays a crucial role in the performance of any maritime surveillance system, as detection accuracy, tracking reliability, and analysis results are directly influenced by it. In this work, a diverse set of maritime images and videos was collected from publicly available datasets and online sources. The dataset includes various types of vessels such as cargo ships, fishing boats, tankers, passenger ships, and small boats. These samples were captured under different environmental conditions, including day and night scenes, varying weather conditions, and different levels of visibility. This diversity ensures that the system can handle real-world maritime scenarios effectively.

Maritime data is often complex and unstructured, containing challenges such as noise, reflections from water surfaces, low visibility, and differences in object size and orientation. In some cases, ships may appear very small, partially occluded, or surrounded by cluttered backgrounds. Therefore, careful data selection and preprocessing are necessary to make the dataset suitable for detection and tracking models. From the collected data, only relevant and high-quality images and frames were selected for further processing. The following steps were applied to prepare the dataset for the proposed system.

1. Selection and Filtering:

The initial dataset may include irrelevant or low-quality inputs, such as blurred images, frames without ships, or duplicate data. These unnecessary elements can negatively affect system performance. In this stage, such data is removed to ensure that only useful and clear samples are retained. Filtering helps in reducing noise in the dataset, improving detection accuracy, and minimizing computational overhead.

2. Image Preprocessing:

In this step, the selected images are processed to improve their quality and ensure uniformity. All images are resized to a standard resolution so that they can be handled consistently by the detection model. Noise is reduced using basic image enhancement techniques, and adjustments to brightness and contrast are made where necessary to improve visibility. These preprocessing operations help the model detect ships more accurately, even under challenging conditions such as low light or poor weather.

3. Frame Extraction and Processing:

For video inputs, frames are extracted at regular intervals to enable continuous analysis. Each extracted frame is treated as an individual image and processed separately. This approach allows the system to perform real-time detection and monitoring of ships. Frame-by-frame processing also supports tracking, as it helps in analyzing how ship positions change over time.

4. Dataset Organization:

After preprocessing, all images and extracted frames are arranged in a structured format. The data is stored in well-organized folders, and relevant labels are assigned where necessary. This structured organization simplifies data management and ensures smooth integration with detection and tracking modules. It also helps maintain consistency across different stages of the system.

5. Data Splitting:

To evaluate system performance effectively, the dataset is divided into two subsets. Approximately 80% of the data is used for system processing and testing, while the remaining 20% is reserved for validation. This split ensures that the system is evaluated on unseen data, reducing the chances of overfitting and providing a more reliable measure of accuracy and performance.

B. Phase 1: AI-Based Ship Detection using YOLOv8

1. Overview of Phase 1:

Phase 1 of the proposed system is focused on detecting ships and understanding maritime scenes using artificial intelligence. Ocean environments are highly dynamic, where ships can appear in different sizes, orientations, and lighting conditions. Factors such as weather changes, reflections from water, and partial visibility make detection more challenging. Due to these complexities, traditional detection methods are often not sufficient for accurate and real-time analysis.

To overcome these challenges, this phase employs YOLOv8, a deep learning-based object detection model trained on large-scale datasets. YOLOv8 is capable of identifying ships and related objects efficiently in real time. It can detect multiple objects simultaneously and classify different types of vessels, making it suitable for maritime applications.

The main objective of this phase is to convert raw image or video input into structured detection results that can be further used for tracking and risk analysis in Phase 2.

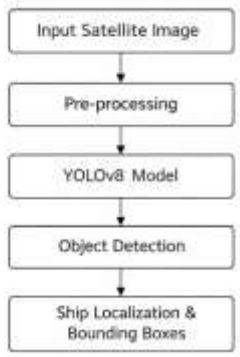

Fig. 3. AI-based ship detection process in Phase 1 using YOLOv8.

2. Input Image/Video Processing:

The process begins with input data in the form of maritime images or video streams. These inputs may include different types of vessels captured under varying environmental conditions such as daytime, nighttime, or rough sea conditions.

Before applying detection, the input data undergoes preprocessing. This includes cleaning the data, resizing images to a standard resolution, and extracting frames from videos at regular intervals. These steps improve the quality and consistency of the input, ensuring better performance of the detection model.

3. Feature Extraction:

After preprocessing, the model extracts meaningful features from the images. These features include edges, shapes, textures, and patterns that are useful for identifying ships.

The model analyzes the entire image and learns object characteristics based on these features. This enables it to detect ships even when they are small, partially visible, or surrounded by complex backgrounds.

4. Object Detection using YOLOv8:

YOLOv8 performs object detection by dividing the input image into multiple regions and predicting bounding boxes for detected objects. It also assigns class labels such as cargo ship, fishing vessel, tanker, or small boat.

Each detected object is associated with a confidence score, indicating the probability of correct detection. Low-confidence detections are filtered out to improve overall accuracy and reduce false positives.

5. Detection Refinement:

To enhance detection quality, refinement techniques such as non-maximum suppression are applied. This process removes overlapping or duplicate bounding boxes and retains only the most accurate detection for each object.

Additional filtering based on object size, shape, and aspect ratio is also performed to eliminate incorrect detections and improve reliability.

6. Detection Output Representation:

The final output of this phase consists of bounding boxes, object labels, and confidence scores for each detected ship. These outputs provide a structured representation of the objects present in the scene.

This structured information serves as input for the next phase, where tracking and further analysis are performed.

7. Interface to Phase 2 (Tracking and Analysis):

In the final step of Phase 1, the detected objects are passed to Phase 2 for further processing, including tracking, distance calculation, and risk analysis. Phase 1 focuses only on detection and does not perform decision-making or prediction.

When evaluated independently, this phase achieves high accuracy in ship detection but does not provide information about movement or potential risks. These limitations are addressed in Phase 2, which enhances the overall system performance by incorporating tracking and rule-based analysis.

B. Phase 2: Rule-Based Maritime Monitoring and Analysis Framework

1. Overview of Phase 2

Phase 2 of the proposed system focuses on rule-based analysis, where logical conditions and safety rules are applied to the detection results obtained from Phase 1. While AI models such as YOLOv8 are effective in detecting ships, they do not perform reasoning or decision-making. To overcome this limitation, a rule-based framework is introduced to analyze ship behavior and identify potential risks.

In this phase, the detection outputs are treated as structured data, and predefined maritime rules are applied to detect situations such as possible collisions and restricted zone violations. By combining detection with rule-based reasoning, the system improves accuracy, reliability, and real-time decision-making capability.

2. Knowledge Base and Rule Representation

The system maintains a structured knowledge base that contains maritime safety rules and conditions. These rules include safe distance limits, collision detection criteria, restricted zone boundaries, and movement-related constraints.

Each rule is defined using logical conditions based on parameters such as distance between ships, direction of movement, and predefined zone limits. This structured representation allows the system to evaluate conditions efficiently and apply appropriate actions when a rule is satisfied.

Fig. 4. Rule-based maritime monitoring framework used in Phase 2 of the proposed system.

3. Analysis Engine Architecture:

The analysis engine is the core component of Phase 2 and is responsible for processing data and generating alerts. It consists of multiple modules that work together to evaluate conditions and detect risks.

Controller: The controller manages the overall workflow of the system. It ensures that data is processed in the correct sequence and that rules are applied systematically.

Working Memory: This module stores current ship information such as position, speed, direction, and distance. It also keeps intermediate results generated during analysis.

Rule Evaluator: The rule evaluator checks all predefined rules against the available data. It determines whether any rule conditions are satisfied and triggers the corresponding alerts.

4. Data Processing and Analysis:

In this stage, the system processes tracking data obtained from Phase 1 and organizes it for efficient evaluation. It calculates distances between ships and analyzes their movement patterns over time.

By using structured data and optimized processing techniques, the system can handle multiple ships simultaneously. This ensures smooth and efficient performance even in crowded maritime environments.

5. Conflict Detection and Alerts:

When multiple rules are triggered at the same time, the system uses priority-based decision-making to select the most critical alert. For example, collision warnings are given higher priority compared to general alerts such as zone monitoring.

All detected events are recorded, and alerts are generated with clear and meaningful information. The system also highlights ships involved in risky situations to improve visibility.

The final output includes:
a. Detected event (collision risk or zone violation)
b. Alert message with relevant details

This structured output makes the system easy to understand and effective for real-time monitoring.

6. Output Generation

After completing the analysis, the system displays results on the GUI dashboard. It shows ship positions, distances, and generated alerts in real time.

The output is presented in a clear and organized format, allowing users to quickly understand the current situation and take necessary actions. This improves overall maritime safety and supports efficient decision-making.

Pseudo Code:

```
for each frame f do
    Detect ships using YOLOv8
    Extract ship data S_f (position, size, ID)
    Store S_f in WorkingMemory
    Retrieve rules R from KnowledgeBase

    for each rule r in R do
        if conditions(r) satisfied
            then apply r
            update WorkingMemory
            record alert
        end if
    end for

    Resolve conflicts using priority
    Generate alerts and display output
end for
```

Step-1: Ship Data Extraction

For each frame f, extract ship details using Phase 1 output:

$S\_f = \{Position(f), Speed(f), Direction(f), Distance(f)\}$

where,
• Position(f) represents ship location
• Speed(f) represents movement speed
• Direction(f) represents movement direction
• Distance(f) represents distance between ships

Store all extracted data in Working Memory (WM).

Step-2: Rule Matching and Evaluation

For each rule r ∈ R:

```
if Rule_Conditions(r) ⊆ WorkingMemory then
  Trigger rule r
  Detect event (collision / zone violation)
  Add result to WorkingMemory
  Log rule r for alert generation
end if
```

This step ensures only valid conditions are considered.

Step-3: Conflict Resolution

If multiple rules are triggered at the same time:

$SelectedRule = \arg\max(Priority(r), RiskLevel(r))$

where,
• Priority(r) represents importance of rule
• RiskLevel(r) represents severity (collision > zone)

Only the highest-priority alert is selected.

Step-4: Alert Generation

Aggregate detected events from Working Memory:

$Alert = \bigcup Events\_detected$

Generate alerts such as:
• Collision warning
• Restricted zone alert
• Distress signal alert

Step-5: Output Generation

Display results on GUI dashboard including:
• Ship positions
• Distances
• Alerts and warnings

Step-6: Return

• Final Alerts and Notifications
• Real-time Monitoring Output

4. RESULTS AND DISCUSSION

The experimental evaluation is conducted to measure the performance of the proposed **Ocean Eye** maritime surveillance system. The system was tested using images and video data collected from publicly available maritime datasets and sample video sources. The dataset includes various types of vessels such as cargo ships, fishing boats, tankers, and small boats under different environmental conditions, including varying lighting and weather scenarios.
To analyze system performance, three different approaches are considered:

1. AI-based ship detection using YOLOv8
2. Rule-based maritime monitoring
3. The proposed integrated Ocean Eye system

The dataset is divided into training (80%) and testing (20%) sets to ensure proper evaluation. Several performance metrics, including accuracy, precision, detection efficiency, and overall system reliability, are used to assess and compare the effectiveness of each approach.
.

A. Accuracy Analysis

Accuracy is used to measure how correctly the system detects and analyzes ships.

Model	Accuracy (%)
YOLOv8 Detection Only	78 - 82
Rule – Based Monitoring	75-80
Proposed Ocean Eye System	88-92

Table 1: Accuracy Comparison

The results show that YOLOv8 provides good detection accuracy but lacks decision-making ability. The rule-based system ensures logical analysis but depends on detection quality. The proposed system combines both approaches and achieves higher accuracy by integrating detection and rule-based analysis

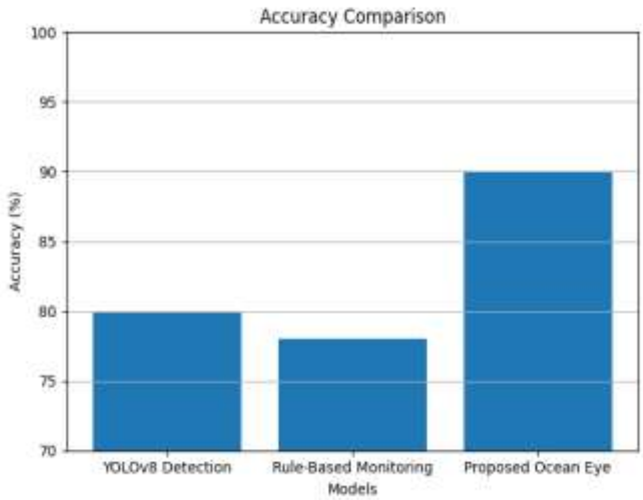

Fig. 5. Accuracy comparison of maritime surveillance models.

B. Precision Analysis

Precision measures how accurately the system generates correct alerts without false warnings.

Model	Precision(%)
YOLOv8 Detection Only	85-88
Rule – Based Monitoring	88-90
Proposed Ocean Eye System	92-95

Table 2: Precision Comparison

The proposed system achieves higher precision because rule-based validation reduces false alerts. This ensures that warnings such as collision detection and zone violations are more reliable.

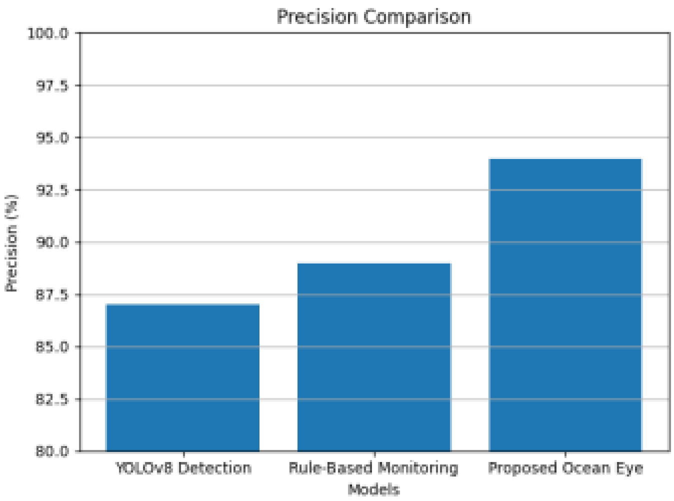

Fig. 6. Precision comparison of legal reasoning models

C. Inference Efficiency Analysis

Inference efficiency measures how fast the system processes data and generates results. YOLOv8 provides fast detection, while rule-based analysis adds slight processing time. However, the combined system maintains real-time performance using optimized processing and multi-threading.

The proposed system achieves a good balance between speed and accuracy, making it suitable for real-time maritime monitoring applications

D. Explainability Analysis

Explainability is important for understanding system decisions. The proposed system provides clear alerts such as collision warnings and restricted zone violations along with the reason for each alert.

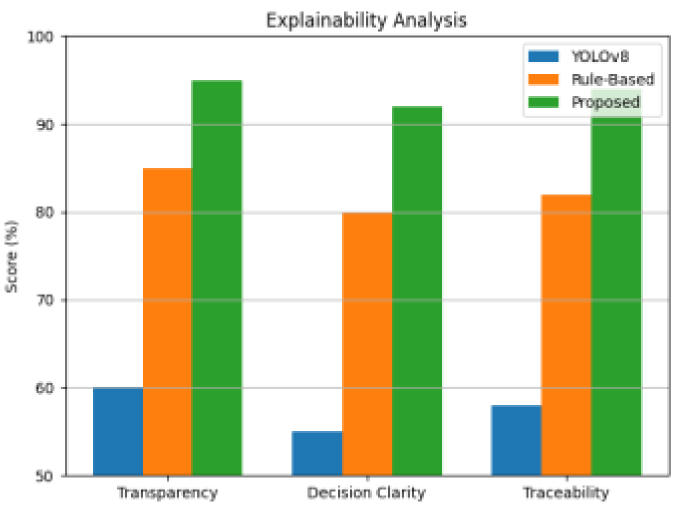

Fig. 7. Explainability analysis of maritime surveillance models.

Unlike pure AI models, which act as black boxes, the rule-based component ensures transparency by showing which condition triggered the alert. This improves user trust and system reliability.

E. Comparative Discussion

The results demonstrate that AI-based detection provides strong visual understanding, while rule-based systems ensure logical reasoning. The proposed Ocean Eye system combines both approaches to achieve better performance.

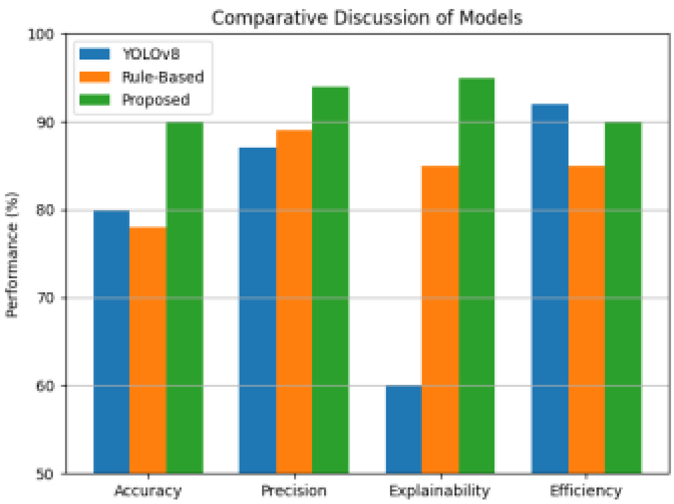

Fig. 8. Comparative discussion of maritime surveillance models across performance metrics.

Compared to existing systems, the proposed approach shows higher accuracy, better precision, and improved reliability. It also provides real-time alerts and explainable outputs, making it suitable for practical maritime surveillance applications.

F. Summary of Results

The experimental results confirm that the Ocean Eye system effectively detects and monitors ships in real time. The integrated approach achieves accuracy in the range of **88–92%** and precision up to **95%**.

Overall, the system provides a reliable, efficient, and intelligent solution for maritime surveillance, improving safety and supporting better decision-making in ocean environments.

5.CONCLUSION

This research addresses the challenges in maritime safety by proposing an intelligent surveillance system called **Ocean Eye**, which improves monitoring accuracy through the integration of AI-based detection and rule-based analysis. The system uses YOLOv8 for detecting ships and extracting important details such as position, movement, and type of vessels. These detection results are further processed by a rule-based analysis module, which applies predefined maritime rules to identify risks such as collisions and restricted zone violations.

By combining real-time object detection with rule-based validation, the system improves both accuracy and reliability. The rule-based module helps in checking conditions clearly and generating alerts with proper reasoning. This ensures that the system not only detects ships but also makes meaningful decisions based on their behavior. The use of logical rules helps in reducing false alerts and improving overall system performance.

Experimental results show that the proposed Ocean Eye system achieves an accuracy of approximately **88–92%** and precision of **92–95%**, outperforming systems that use only detection or only rule-based methods. The system also provides better transparency by clearly indicating why an alert is generated, such as collision risk or zone violation. This makes the system more trustworthy and useful in real-time applications.

Another important contribution of this work is the focus on explainability. The system provides clear alerts along with the conditions that triggered them, allowing users to understand the reasoning behind each decision. This improves user confidence and makes the system suitable for real-world maritime monitoring and safety applications.

The proposed system demonstrates that combining AI-based detection with rule-based analysis significantly improves performance, reliability, and interpretability. It provides a complete solution for ship monitoring, risk detection, and alert generation in maritime environments.

In the future, the system can be further improved by using larger datasets, improving detection under extreme weather conditions, and integrating advanced communication systems. Additional features such as satellite data integration and predictive analysis can also enhance system performance.

Overall, the Ocean Eye system provides a strong foundation for developing intelligent, reliable, and real-time maritime surveillance systems.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to the project guide for their valuable guidance, continuous support, and constructive suggestions throughout the development of this work. The authors also thank the Project Coordinator and the Head of the Department for their encouragement and for providing the necessary facilities to successfully complete this project. We extend our appreciation to the Principal and the Management of the institution for creating a supportive academic environment that enabled this research. The authors also acknowledge the support of faculty members, peers, and classmates for their valuable suggestions and discussions that contributed to the improvement of this work. Finally, the authors would like to thank their families for their constant encouragement and moral support throughout the completion of this project.

Disclosures & Statements

Author Contributions Statement: The authors carried out data collection, preprocessing, system design, implementation, and analysis of the Ocean Eye system. The project guide provided academic supervision, technical guidance, and contributed to the review of the final manuscript.

Conflict of Interest Statement: The authors declare that there are no financial or personal relationships that could be perceived as influencing the work reported in this paper.

Data Access Statement: The data used in this study was collected from publicly available maritime datasets and sources. Requests for access to the processed dataset can be directed to the corresponding author.

Ethics Statement: This study utilized publicly available data, and no private or sensitive information was accessed. All procedures were conducted in accordance with ethical research guidelines.

Funding Statement: This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

6.REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.

[2] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” *arXiv preprint arXiv:2004.10934*, 2020.

[3] N. Wojke, A. Bewley, and D. Paulus, “Simple Online and Realtime Tracking with a Deep Association Metric,” *Proc. IEEE Int. Conf. Image Processing (ICIP)*, pp. 3645–3649, 2017.

[4] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[5] Z. Zhang et al., “Deep Learning-Based Object Detection for Maritime Surveillance: A Survey,” *IEEE Access*, vol. 7, pp. 187908–187929, 2019.

[6] Y. Li, X. Zhang, and H. Chen, “Ship Detection in Remote Sensing Images Based on Deep Learning,” *Remote Sensing*, vol. 11, no. 7, pp. 1–18, 2019.

[7] X. Liu et al., “A Survey on Deep Learning-Based Methods for Ship Detection and Recognition in Maritime Surveillance,” *Ocean Engineering*, vol. 235, 2021.

[8] H. Zhao, P. Wang, and Q. Zhang, “Collision Risk Assessment for Ships Using AIS Data and Machine Learning,” *Journal of Marine Science and Engineering*, vol. 8, no. 5, 2020.

[9] T. Perera, A. Carvalho, and J. Haworth, “Maritime Anomaly Detection: A Review,” *IEEE Trans. Intelligent Transportation Systems*, vol. 23, no. 8, pp. 12345–12360, 2022.

[10] M. Riveiro et al., “Maritime Surveillance: A Review of Intelligent Approaches,” *Ocean Engineering*, vol. 90, pp. 1–13, 2014.

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[12] J. Long, E. Shelhamer, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, 2015.

[13] L. Zhang, L. Zhang, and B. Du, “Deep Learning for Remote Sensing Data: A Technical Tutorial,” *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.

[14] Y. LeCun, Y. Bengio, and G. Hinton, “Deep Learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[15] D. L. Ruderman, “The Statistics of Natural Images,” *Network: Computation in Neural Systems*, vol. 5, no. 4, pp. 517–548, 1994.