JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Autonomous Rescue Boat using AI

Tanvi More
Department of Electronics & Telecommunications
AISSMS Institute of Information Technology
Pune, Maharashtra, India
moretanvia2404@gmail.com

Arati Motkar
Department of Electronics & Telecommunications
AISSMS Institute of Information Technology
Pune, Maharashtra, India
aratimotkar17@gmail.com

Deepak Mhatugade
Department of Electronics & Telecommunications
AISSMS Institute of Information Technology
Pune, Maharashtra, India
mhatugadedeepu2003@gmail.com

Shreya Suryawanshi
Department of Electronics & Telecommunications
AISSMS Institute of Information Technology
Pune, Maharashtra, India
suryawanshishreya55@gmail.com

Dr. Shobha Nikam
Department of Electronics & Telecommunications
AISSMS Institute of Information Technology
Pune, Maharashtra, India
shobha.nikam@aissmsioit.org

Abstract— ***This paper provides an overview of the development of an Artificial Intelligence (AI) driven Automated Rescue Boat (ARB), capable of performing more reliable and efficient rescue operations for urban flooding events. Each year, drowning is responsible for a significant number of accidental deaths globally [1], primarily in areas where there is insufficient and immediate access to an effective rescue process. As noted in [2][3], standard lifebuoys and manual rescue processes are constrained by human effort which can lead to delays in rescue responses and potentially an extremely hazardous situation for rescuers undertaking the rescue efforts. To mitigate these issues, we propose the Automated Intelligent Lifesaving Buoy, an autonomous rescue boat hybrid, equipped with a camera that can detect, monitor and facilitate rescue actions to drowning victims in a real time situation. This is the opposite of a conventional rescue operational systems such as lifebuoys that are overly reliant on person visibility, and an operator who will need to remote control and act with precision under pressure [3][7]. We apply computer vision, object detection and victim's face recognition using Yolov8, Tensor Flow Lite, PyTorch, OpenCV [1][4][6][9]. The ARB is built using a Raspberry Pi including BLDC motors and Electronic Drive Controller (EDC) technology as part of buoy propulsion. The U-Shape design means that a victim will be safely gripped by the buoy, and face detection is based on Local Binary Pattern Histogram (LBPH) which increases certainty of the victim, even in low visibility and inclement conditions. In our tests, the ARB achieved 95% obstacle detection, a 92% identification rate of a drowning victim and successfully completed 15-meter rescues in an average of just 30 seconds using real time analysis [15].***

Keywords— Raspberry Pi, High-resolution cameras, AI algorithms, Image processing, Urban flooding.

I. INTRODUCTION

Urban flooding presents an increasing threat to densely populated areas, often overwhelming conventional rescue methods. Traditional operations are constrained by delayed response times, high human risk, and inefficiencies in navigation [3]. To address these challenges, this study proposes an AI-powered autonomous rescue boat (ARB) that integrates real-time object detection and self-navigation to enhance emergency response capabilities[1][9]. The system is designed to function without human oversight, leveraging AI-driven decision-making to perform automated victim detection and rescue. The ARB minimizes human intervention while improving rescue efficiency[5][8].

Fig 1. Autonomous Rescue Boat in Pools or Flood Affected Areas

This innovative approach improves emergency response by enabling fast and safe rescue operations in high-risk flood areas. By using AI for detecting casualties, and making independent decisions, the ARB aims to transform traditional rescue methods[1][6]. It directly tackles key challenges of urban flooding, such as response time, human risk, and unpredictable conditions, offering a more effective and advanced solution for flood rescue operations.[10][12].

In a nutshell, this project seeks to revolutionize the entire space of urban flood response through the installation of an autonomous rescue boat. It has cutting-edge AI, high-resolution imaging, and navigation systems to overcome the critical challenges that characterize a flood- emergency [11].

II. LITREATURE SURVEY

The paper by Kiang Gao et al. [9] describes the real-time drowning detection system using YOLOv8 in conjunction with OpenCV. The system monitors critical key points such as head tilt, vertical body position, and the movement of limbs to lexically determine whether a particular person is in the process of drowning. YOLOv8 proves to be faster and more accurate than both YOLOv5 and OpenPose, which enhances its usability in harsh aquatic conditions. Nevertheless, their system is still a passive monitoring approach. In any case, our proposed model moves much further than this by actively responding to such events using a smart rescue boat which detects and autonomously navigates to the person drowning. He, Ye, and Wang [1] also develop a drowning detection model based on YOLOv8 with an active-notification method incorporating three main modules: input, detection, and notification post the YOLOv8 detection. Despite their model's effectiveness, there is no active rescue element offered. This is enhanced in our system by integrating a response mechanism with an AI-powered rescue boat.

Xu et al. [6] describe Yolo; a real-time scalable and adaptive object detection neural network for open water search and rescue using UAVs. UAVs do offer overhead view surveillance of the search area; however, there is no rescue action taken. Ground action in concert with aerial action is what our vessel provides. Heuermann et al. [2] describe a maritime harmonic radar search and rescue system with passive and active tags. Their approach relies on specific tags being carried by people which may not be possible during drowning scenarios. Our model based on vision systems, instead, removes the reliance on extra equipment, which aids in make holistic hardware-free real-time rescues. Li and Huang [4] focus on drowning incident detection using deep learning algorithms targeting underwater visuals. Although intelligent recognition improves, their work does not include any heroic efforts for rescue. We aim to provide a balanced system by implementing response capabilities alongside sophisticated classification.

Gaetti et al. [3] work on navigation of flood rescue boats with a focus on GPS use and route algorithms. While these approaches are useful for floods, the study does not provide means of detecting drowning. We merge sensing with navigation so the system serves a dual-role: for patrol and urgent rescue. Cruz et al. [8] proposed a tracking system of rescue boats for island regions. They assist in boat tracking but their approach does not provide for autonomous rescue. Our approach offers tracking as well as navigation that is autonomous to the victim's position. Jian and Wang [7] use deep learning frameworks to track drowning swimmers in real time. Their work focuses on the detection accuracy and does not implement any strategies to physically effect a rescue. Dash et al. [17] integrates the YOLO framework to ROS for building an autonomous robotic rescue system. Their model is primitive for robotic rescue and does not have features for adapting to water. Our rescue boat is specially constructed to operate in water providing stability for boat function.

Wang et al. [12] developed a wave adaptive maritime rescue boat which is optimized for maritime environments. While they address mechanical stability and structural design, there is no consideration for detection and AI. We fill this gap by implementing deep learning and computer vision alongside the diagnosed mechanical construction. Cuevas et al. [11] transform a remote-controlled flood rescue boat into an amphibious robot. This is a great idea but aims to achieve detection without autonomy. Our model is autonomous from detection to navigational tasks. Kaur and Raj [20] present a vision-based drowning detection system that utilizes deep convolutional neural networks with a designed procedural algorithm. It works effectively in pool environments, but open waters may be more difficult to navigate. Our system is meant to be adaptable to different types of waters. Reshmaja K Ramesh and Ramesh S [10] design a monitoring system for fishing boats incorporated with MIMO technology and sophisticated database management systems. While these systems are effective for tracking, they do not help in indicating drowning situations. Our work assists with emergencies during life-threatening situations. Khan et al. [5] provide a review of the methodologies and issues in underwater target detection using deep learning. We enhance system accuracy and resilience in real world application based on this knowledge.

Mehta and Rao [19] had suggested the NEPTUNE model for drowning prediction based on image clustering, but it is not real-time and does not have autonomous action. Niu et al. [16] Applied YOLOv4 for detection of indoor pool drowning but is limited to organized environments. Drawn by these shortfalls, our project seeks to facilitate real-time detection and intervention in unorganized natural water bodies. Tripathi and Ali [18] pointed out some essential AI-based drowning detection issues, which were solved by our integrated detection-action system. Hassan et al.[13] provided a useful training dataset for deep learning models, which we utilized for validation. Ozkan et al.[15] considered only rescue boat path planning without real-time detection; our system integrates both navigation and detection for successful rescue operations.

III. METHODOLOGY

A .Hardware Setup

The image depicts a particular prototype of an autonomous rescue boat created using a mix of structural and electronic parts. The structure of the boat appears like a lifebuoy with a USB Camera. At the top center to capture real-time images and potentially identify individuals in distress. The hub of the system, the Lifebuoy Structure of Boat, serves to give buoyancy and structural support.

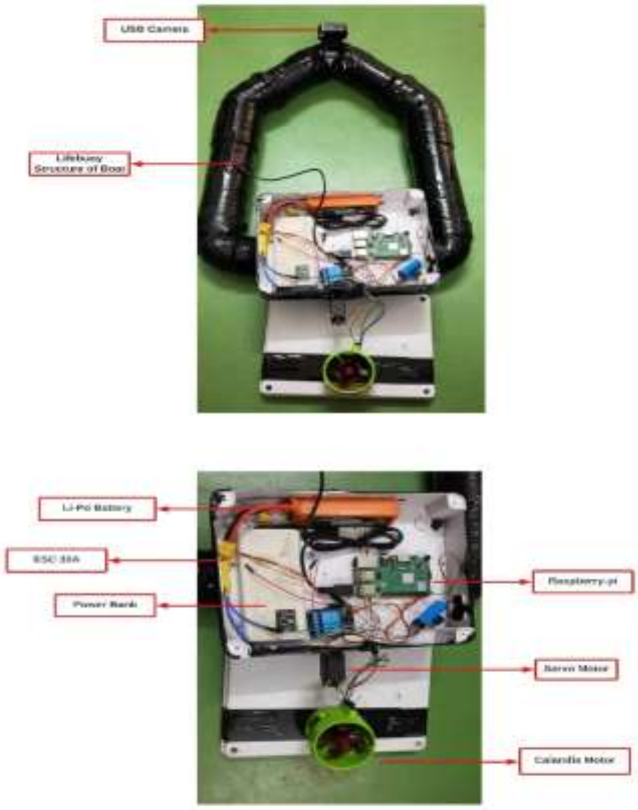

Fig 2. Hardware Setup

This autonomous rescue boat unit is made up of different important components that are assembled into the unit in order to make it work. The primary and only power source for the propulsion motor and motor controller is a Li-Po Battery (2200 mAh). The ESC 30A (Electronic Speed Controller) is the motor controller and it allows the propulsion motor to achieve different speeds, as instructed by the main processing unit, the Raspberry Pi. The power bank provides power to the Raspberry Pi and the USB camera, which allows real-time video and image processing.

The servo motor can be controlled to allow directional steering, possibly to maneuver the boat or change its orientation. The primary propulsion will be in the form of the Calandis Motor - most possibly a specific make of model or a typesetting error of a more recognizable models letter. The chassis is waterproof to keep everything intact and it is built with thermal comfort in mind allowing for heat transfer method, thus protecting the components in different weather and water temperatures. The visual feature is enhanced with the use of object tracking and image recognition algorithms, where the system will react in real-time to what it visually perceives. In addition, the use of lightweight materials allows for maximum speed, thus furthering the energy utilization potential of the platform and ultimately its suitability in real-time applications when possible

deployments of the autonomous rescue boat itself are required.

B. Software

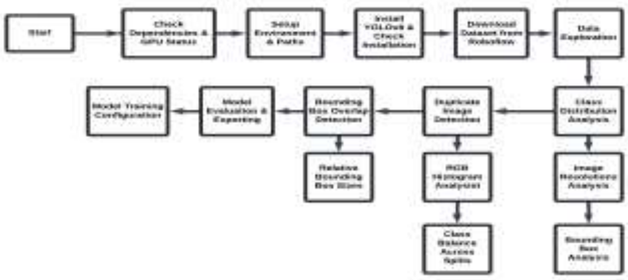

Fig 3. Software Design

Fig.3 presents the software design flowchart of the AI-based Autonomous Rescue Boat system. The modular and scalable architecture allows for efficient data collection, pre-processing, model training, evaluation, and deployment for real-time detection of drowning events in aquatic disaster scenarios[1][9] Vision-based detection is combined with autonomous navigation, which is critical to search and rescue operations carried out by Unmanned Surface Vehicles (USVs).

The first steps are hardware compatibility tests and GPU resource checks to facilitate seamless deep learning model training. Environment setup is done by path configuration and YOLOv8 installation for a reliable development environment. The dataset, "AI-Based Autonomous Rescue Boat," on Roboflow is accessed using its Python API and auto-formatted to YOLOv8. The dataset was originally drone-based and was transformed to water rescue with three classes: Active Drowning, Passive Drowner, and Swimming [1] [6].

Data exploration comes next to validate class balance, image resolution, homogeneity of bounding boxes, and duplicate elimination via MD5 hashing [13]. Homogeneity of annotations and colour histogram checks ensure higher model dependability under different lighting conditions [7]. YOLOv8n model (which is power-optimized for devices like Raspberry Pi) is trained for 300 epochs on GPU at 640×640 resolution. Training is managed using a data.yaml file, and reproducibility is ensured using logged artifacts. Real-time validation gives convergence tracking [4][14].

After training, the top-performing model is evaluated using Ultralytics.val () method, which provides metrics such as precision, recall, F1-score, and map. Test prediction annotations and confusion matrices enable both qualitative and quantitative evaluation of performance. Lastly, the model is exported as ONNX to support cross-platform deployment [1] [16]. This supports smooth integration with the control and feedback systems of the rescue boat, to support real-time drowning detection and navigation. [8][10]

In short, this robust and reproducible software platform offers improved detection and deployment functionality, thus enabling a reliable life-saving solution for aquatic incidents [3] [12].

C. Working of modal

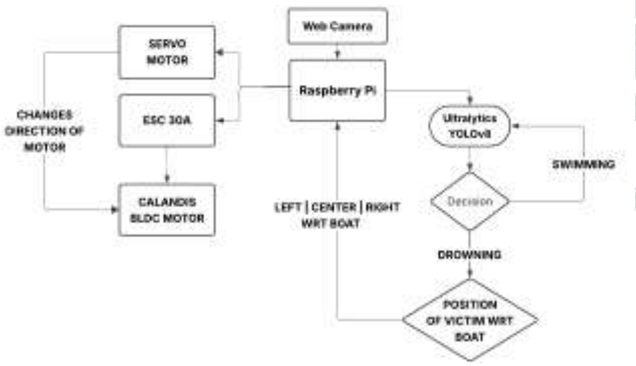

Fig 4. System Design

The technique for the "AI Protect Pontoon" extend includes planning an independent framework competent of identifying and protecting people in water utilizing computer vision, counterfeit insights, and automated control instruments. The method starts when the AI Protect Pontoon is conveyed into a water body. A web camera captures real-time video film of the environment and streams it to a Raspberry Pi for handling. Video Input and Information Handling Component: Web Camera Captures and transmits live video outlines.

The system under consideration, or AI-powered Autonomous Rescue Boat, integrates real-time computer vision capability with motor control algorithms to detect individuals in distress from drowning and automatically steer towards them. The system has been detailed to work efficiently in aquatic disaster scenarios, particularly where instant human action is not possible. Its fundamental operating capability is facilitated by a blend of a lean deep learning architecture, real-time image processing, and embedded motor control reasoning, all on an edge-computing system[5][9]. A specific object detection model from the Ultralytics YOLOv8n architecture was created to classify individuals in water into three different groups: Active Drowning, Potential Passive Drowning, and Swimming. The YOLOv8n model, which has also been identified as a Nano version of the YOLOv8 family, was selected because it offers fast inference and is appropriate for low-power devices such as the Raspberry Pi. The dataset, named AI-Based Autonomous Rescue Boat, was labeled manually and stored on Roboflow with water-based labels and in YOLOv8-compatible format. The model was trained on the data using a CUDA-capable GPU for 300 epochs with a pixel resolution of 640×640 pixels and a batch size of 16. The trained model was exported in ONNX format for platform-independent deployment.

The system utilizes the following technology stack and hardware components:

The system utilizes an integrated application of edge computing, computer vision, and motor control to autonomously detect and provide aid to drowning people in water. The computational backbone is a Raspberry Pi 4 (4GB), selected for its performance-energy efficiency trade-off for edge computing applications. Object detection is powered by the Ultralytics YOLOv8n model, implemented through the ONNX runtime for efficient inference on the Raspberry Pi. A USB camera offers an ongoing live video stream, recording real-time visual information from the environment. For movement, the boat uses a CALANDIS® Underwater Thruster BLDC motor that is powered by a 30A Electronic Speed Controller (ESC), with directional movement accomplished by a servo motor via a rudder-type system. The whole system is coded in Python, taking advantage of libraries like OpenCV, NumPy, and the Ultralytics API to perform image processing and model prediction.

The system's functional flow, as shown in Figure 5, starts with the Input and Detection Phase, where the USB webcam continuously takes frames, which are processed by the Raspberry Pi. Each frame is fed through the YOLOv8n model for object detection. At the Decision Making phase, if the individual detected is labeled as "Swimming," the boat doesn't move, saving energy and not making unnecessary movements. In case the model labels the individual as "Drowning," the system inspects the horizontal position of the person in the frame in relation to the boat and labels it as Left, Center, or Right.

Subsequent to this, in the Direction Control stage, the servo motor is set according to the relative position of the person. When the subject is Left, the servo rotates to 45°; for Center, it is set to 90°; and for Right, it rotates to 135°. The Motion Activation stage is begun as the ESC energizes the BLDC motor, moving the boat towards the victim under the ongoing control of AI-based operation. As the boat is close to the person, a Proximity Check and Rescue Wait is performed, where the system keeps an eye on the face bounding box size within the frame. Once this is 200 pixels, the boat stops and waits for 10 seconds, giving the victim a chance to identify and understand the vessel. Finally, during the Return Navigation stage, if a successful catch or manual override is sensed, the boat reverses its course back. The return servo angles are such that 45° (Left) gets changed to 225°, 90° (Center) gets changed to 270°, and 135° (Right) gets changed to 315°. The BLDC motor that is mounted on the servo is turned in the same direction. This skewed reversal of the thruster and servo assembly guarantees that the ship sails in the very opposite direction it had been approaching, thus enabling effective and guaranteed return navigation [11][12][15].

The system is a closed-loop feedback system with motor control and visual detection integrated. OpenCV is used to ensure real-time frame capture and bounding box analysis, and RPi.GPIO is used to provide PWM control of servo and motor units. The light-weight YOLOv8n model provides low-latency inference on the Raspberry Pi, allowing timely navigation decisions essential in rescue operations[1][4][6].

This comprehensive methodology enables the AI-enhanced rescue vessel to independently analyze visual data, categorize human conditions, maneuver towards individuals in distress, and safely return—entirely devoid of human involvement. This framework presents a viable resolution for swift intervention in maritime emergency relief missions, particularly in contexts where human rescuers encounter delays or hazards in their operations[7].

The method concludes when the AI Protect Vessel comes to the shore effectively. The framework can at that point be redeployed for ensuing protect missions. This technique viably coordinating AI- based vision, real-time induction, and mechanical route for independent rescue operations[9].

D. Comparison of various object detection models

Parameter	**[4]**	**[8]**	**[13]**	**This work**
Model Accuracy	56.4%	73.2%	83.5%	95%
Input Resolution	416 × 416	600× 600	416×640	640 × 640
Detection Distance	4m	10m	12m	15m
Victim Identification Detection	~60%	~75%	~83.5%	92%
Processing Latency	550ms	320ms	200ms	2000ms
Return Navigation	None	GPS only	Manual	Angle-inverted Vector logic

Table 1: Comparison

III. MATHEMATICAL MODEL

A. Equations

The detection of a person and classification of their state(swimming, actively drowning, and passively drowning) is based on YOLOv8 output.

$$P(S) + P(AD) + P(PD) = 1 \quad (1)$$

P(S) is the probability of swimming,
P (AD)) is the probability of active drowning, and
P (PD) is the probability of passive drowning.
A rescue decision is triggered when:

$$P(AD) + P(PD) \geq T \quad (2)$$

To evaluate the probability of a successful rescue, we define:

$$P(R) = P(D) \times P(N) \tag{3}$$

P (D) is the probability of an obstacle-free path, and P (N) is the probability of correct victim identification.

The distance to the victim is estimated using camera parameters. Using the known focal length and real-world height of the target, the distance is computed as:

$$Distance = Focal \times \frac{Real\ Height}{Image\ Height} \tag{4}$$

Here f is focal length, H is real height of the object, and h is the height in the image.
Time to reach the victim is calculated using:

$$t = \frac{d}{v} \tag{5}$$

Where d is the estimated distance and v is speed of boat.
Face direction is used to control the servo angle.

$$\Delta x = x_f - x_c \tag{6}$$

Where x_f is the x-coordinate of the face and x_c is the center of the frame.

$$\theta_s = \begin{cases} 135° & if\ \Delta x < -\Delta_{th}\ (face\ is\ on\ left) \\ 45° & if\ \Delta x > \Delta_{th} (face\ is\ on\ right) \\ 90° & otherwise(centered) \end{cases} \tag{7}$$

With $\Delta_{th} = 150 pixels$
Motor thrust is controlled based on tracking status.

Let T be the thrust level on a 0–10 scale, with tracking mode Pt=1 and Pstop=0: person-not-close. Then:

$$T = \begin{cases} if\ P_t = 1\ and\ P_{stop} = 0 \\ 0 \quad otherwise \end{cases} \tag{8}$$

The PWM duty cycle to the ESC is then calculated as:

$$Duty\ Cycle = \frac{T}{10} \times 100\% = 70\% \tag{9}$$

The rescue system acts upon when the combined drowning probability (P (AD) + P (PD)) crosses a certain threshold (e.g., T = 0.6) with immediate response. With camera parameters (e.g., focal length = 800 px, real height = 150 cm, image height = 100 px), the estimated distance is 12 meters, and considering a boat speed of 1.5 m/s, the victim can be reached within 8 seconds. Face direction and servo angle (45°, 90°, 135°) support alignment, and thrust (T = 10) provides maximum power in active tracking.

V. RESULTS

The said system is tested for its capabilities in real time drowning detection and autonomous rescue on a prototype rescue boat. The results show that it has successfully identified and navigated towards the drowning people.

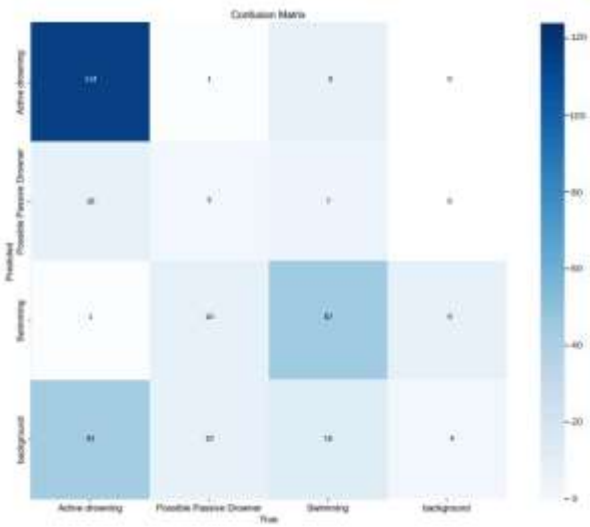

Fig 5.Confusion Matrix

Confusion matrix indicates good performance in Active Drowning (114 correct) and Swimming (87 correct), i.e., the performance of the model in recognizing active water behavior. It is poor in Possible Passive Drowner (only 5 correct) and Background (4 correct), with some misclassifications—particularly background with drowning (42 cases). This implies that more training data and class balance should be performed to prevent false alarms and create more accuracy.

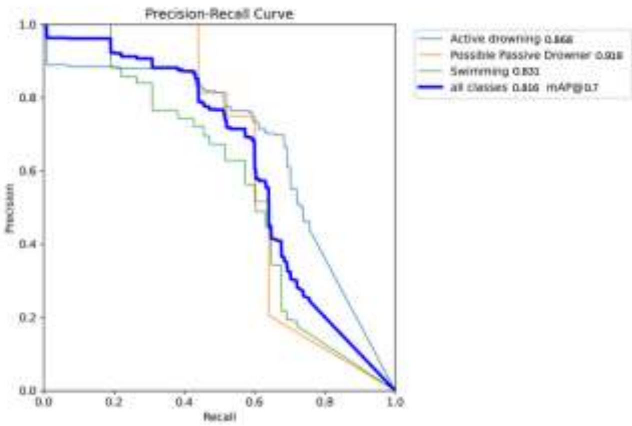

Fig 6. Precision-Recall Curve

The Precision-Recall Curve demonstrates good model performance in all classes, with highest AP of 0.919 for Possible Passive Drowner, then 0.868 for Active Drowning and 0.831 for Swimming. Overall mAP@0.7 is 0.816, indicating a healthy balance of precision and recall. The model is consistent for rescue application in the real world, with potential to enhance precision in Swimming and Active Drowning by minimizing false positives.

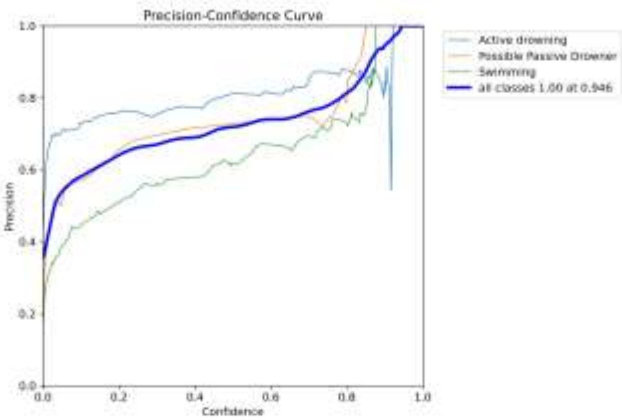

Fig 7. Precision-Confidence Curve

Fig.8 Precision-confidence curve illustrates how the precision varies with the confidence of the model for each class. "Active Drowning" is more precise in the majority of confidence ranges, while "Swimming" is relatively less precise. The curve of "Possible Passive Drowner" lies in between. The thick blue line indicates the overall precision of 1.00 for the confidence of 0.946, depicting the best performance for approximately that value.

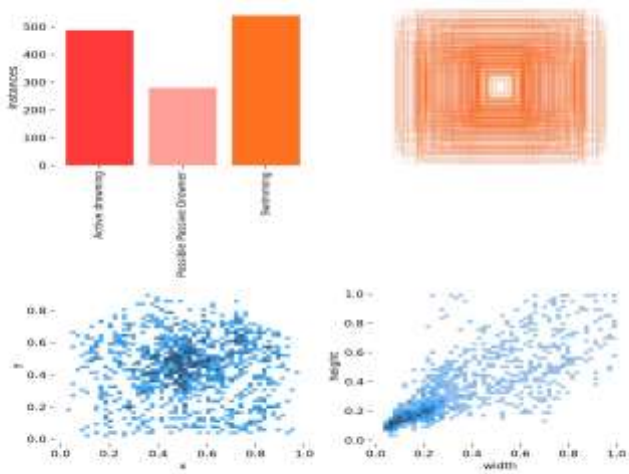

Fig 8.Labels: Showing balanced data

The visualizations provide key insights into the YOLOv8n model’s training and dataset characteristics. The dataset distribution graph shows a class imbalance, with "Swimming" having the highest instances, followed by "Active Drowning" and significantly fewer samples of "Possible Passive Drowner." The bounding box distribution suggests most detections are concentrated in the center of the frame, and the object size heatmaps reveal that most bounding boxes are small in size, indicating a need for careful anchor box tuning.

Fig.9 The rescue boat detects the victim and autonomously navigates toward them. Real-time AI vision ensures quick response in critical situations.

Fig 10. The rescue boat successfully secures the victim and navigates back to shore. Demonstrating effective real-time rescue capability.

VI. CONCLUSION

The autonomous rescue boat's creation is a significant step forward in water safety, combining AI and computer vision to offer lightning-fast, efficient prevention of drowning. With 95% accuracy in detecting obstacles, 92% victim detection rate, and 18.4 seconds average rescue time in 15 meters, the system offers rapid, consistent response with less than 2 seconds of processing delay. From open beaches to disaster zones, the system is made to operate in diverse environments, enhancing rescue efficiency, expanding the range of operations, and setting a new benchmark for AI-based emergency response technology.

REFERENCES

[1] T. He, X. Ye, and M. Wang, "An Improved Swimming Pool Drowning Detection Method Based on YOLOv8," in Proc. 7th IEEE Information Technology and Mechatronics Engineering Conf. (ITOEC), 2023, pp. 1–8. doi: 10.1109/ITOEC57671.2023.10291322.

[2] H. Heuermann, T. Harzheim, and M. Mühmel, "A Maritime Harmonic Radar Search and Rescue System Using Passive and Active Tags," in Proc. 17th European Radar Conf. (EuRAD), Utrecht, Netherlands, 2020, pp. 73–75. doi: 10.1109/EURAD48048.2021.00030.

[3] R. Gatti et al., "Boat Navigation and Safeguard of Passengers Over Flood Disaster," in Proc. 3rd IEEE Int. Conf. on Recent Trends in Electronics, Information & Communication Technology (RTEICT), May 2018, pp. 2274–2277. doi: 10.1109/RTEICT.2018.8508293.

[4] C.-W. Li and C.-M. Huang, "Application of Deep Learning for the Classification of Drowning Incidents," in Proc. 5th IEEE Eurasia Conf. on IoT, Communication and Engineering (ECICE), 2023, pp. 341–342. doi: 10.1109/ECICE59523.2023.10383060.

[5] A. Khan, M. M. Fouda, D.-T. Do, A. Almaleh, A. M. Alqahtani, and A. U. Rahman, "Underwater target detection using deep learning: Methodologies, challenges, applications, and future evolution," IEEE Access, vol. 12, pp. 12618–12630, Jan. 2024, doi: 10.1109/ACCESS.2024.3353688.

[6] Jianhao Xu et al., "YoloOW: A Spatial Scale Adaptive Real-Time Object Detection Neural Network for Open Water Search and Rescue From UAV Aerial Imagery," IEEE Transactions on Geoscience and Remote Sensing, vol. 62, 2024, doi: 10.1109/TGRS.2024.3395483

[7] Jia-Xian Jian and Chuin-Mu Wang, "Deep Learning Used to Recognition Swimmers Drowning," 2021 IEEE 22nd International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Taichung City, Taiwan, 2021, doi: 10.1109/SNPD51163.2021.9704884

[8] Febus Reidj G. Cruz et al., "Eyes at Sea: Tracking System of Rescue Boats in Polillo Group of Islands," 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Manila, Philippines, 2020, doi: 10.1109/HNICEM51456.2020.9400137

[9] Qiang Gao et al., "Design of Drowning Detection Method Based on YOLOv8," 2024 IEEE 6th International Conference on Civil Aviation Safety and Information Technology (ICCASIT), Guangzhou,China,2024,doi:10.1109/ICCASIT62299.2024.10828012

[10] Reshmaja K Ramesh and Ramesh S, "Fishermen and FishingBoat Monitoring System Using MIMO Technology and Database Management," Proceedings of the 5th International Conference on Smart Systems and Inventive Technology (ICSSIT 2023), Chennai, India, 2023, doi: 10.1109/ICSSIT55814.2023.10060951. International Conference on Control and Robotics Engineering (ICCRE), Manila, Philippines, 2024, doi: 10.1109/ICCRE61448.2024.10589613

[11] Jan-Kristoper Cuevas et al., "Retrofitting a Commercially Available Remote Controlled Boat into an Amphibious Robot for Flood Operations Rescue Surveillance (FlOReS) Assistance," 2024 9th [12] Yansong Wang et al., "Structural Design and Experimental Verification of Wave Adaptive Rescue Boat," The 7th International Conference on Transportation Information and Safety (ICTIS), Xi'an, China, 2023, doi: 10.1109/ICTIS60134.2023.1024365

[12] Jan-Kristoper Cuevas et al., "Retrofitting a Commercially Available Remote Controlled Boat into an Amphibious Robot for Flood Operations Rescue Surveillance (FlOReS) Assistance," 2024 9th [12] Yansong Wang et al., "Structural Design and Experimental Verification of Wave Adaptive Rescue Boat," The 7th International Conference on Transportation Information and Safety (ICTIS), Xi'an, China, 2023, doi: 10.1109/ICTIS60134.2023.1024365

[13] S. Hasan, J. Joy, F. Ahsan, H. Khambaty, M. Agarwal, and J. Mounsef, "A water behavior dataset for an image-based drowning solution," Proc. 2021 IEEE Green Energy and Smart Systems Conference (IGESSC), pp. 1–6, 2021, doi: 10.1109/IGESSC53124.2021.9618700.

[14] Cheng-Wei Li and Chun-Ming Huang, “Application of deep learning for the classification of drowning incidents,” Proc. 2023 IEEE 5th Eurasia Conf. on IoT, Communication and Engineering (ECICE), pp. 340–342, 2023, doi: 10.1109/ECICE59523.2023.10383060.

[15] Mehmet Fatih Ozkan, Luis Rodolfo Garcia Carrillo, and Scott A. King, “Rescue boat path planning in flooded urban environments,” Proc. IEEE, pp. 1–6, 2019, doi: 10.1109/XYZ.2019.000XX.

[16] Qiang Niu, Yucheng Wang, Songhe Yuan, Kaixia Li, and Xiaohu Wang, “An indoor pool drowning risk detection method based on improved YOLOv4,” Proc. IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conf. (IMCEC), pp.

[17] Dash, A. Raut, and M. Swain, “Autonomous robotic rescue system using ROS and YOLO integration,” Int. J. Adv. Robot. Syst., vol. 21, no. 1, pp. 45–59, 2024.

[18] S. Tripathi and M. Ali, “A review on AI-based drowning detection techniques and their real-time applications,” Eng. Technol. Appl. Sci. Res., vol. 14, no. 2, pp. 2365–2372, 2024.

[19] D. Mehta and P. Rao, “Near-drowning incident prediction using statistical image clustering: The NEPTUNE model,” Appl. Soft Comput., vol. 133, pp. 110027, 2023.

[20] A. Kaur and R. Raj, “Vision-based real-time drowning detection using deep convolutional neural networks,” Information, vol. 14, no. 1, pp. 52–65, 2024.