A Lightweight Drowning Person Detection Using Deep Learning Algorithm

Ni Made Shavitri Mustikayani[1], Dayen Manoppo[2], Marsel Marhaen Wungow[3], Wahyuni Fithratul Zalmi[4], Muhamad Dwisnanto Putro[5]

Bachelor Program of Informatics, Department of Electrical Engineering, Sam Ratulangi University, Manado, Indonesia[1,2,3,4]

Master Program of Informatics, Postgraduate Program, Sam Ratulangi University Manado, Indonesia[5]

nimustikayani026@student.unsrat.ac.id[1], dayenmanoppo026@student.unsrat.ac.id[2], marselwungow026@student.unsrat.ac.id[3], wahyuni.fithratul.zalmi@unsrat.ac.id[4], dwisnantoputro@unsrat.ac.id[5]

Abstract – Visual obstructions and fatigue often hinder human surveillance in preventing drowning incidents. Manual surveillance methods are susceptible to visual obstructions, distractions from crowds, and fatigue. To address these issues, this study proposes an automated real-time drowning detection system that utilizes state-of-the-art deep learning techniques. We use the YOLOv12-Nano architecture, selected for its balance between detection accuracy and computational efficiency. This model was trained and evaluated on the SelfMade dataset, which covers various water conditions and poses indicating swimmers in distress. In testing, YOLOv12-Nano achieved a mAP@50 of 0.984 and a mAP@50-95 of 0.732, with 2.52 million parameters and a computational requirement of 6 GFLOPs. These results demonstrate that YOLOv12-Nano-based automatic detection provides reliable, resource-efficient real-time monitoring, is suitable for implementation on real-world application, and can support human surveillance and accelerate emergency responses to reduce fatal drowning accidents.

Keywords: Drowning, Object Detection, Convolutional Neural Network, YOLOv12, Deep Learning

I. INTRODUCTION

Water safety faces significant challenges due to unpredictable water dynamics and the vastness of the monitoring area [1]. Drowning incidents often occur silently and rapidly, making them prone to being overlooked without a reliable surveillance system [2], [3]. Response speed is the most critical factor; every second of delay in detecting a victim in distress significantly reduces their chances of survival [4]. Therefore, integrating smart monitoring technology capable of real-time detection is essential to strengthen public safety infrastructure in high-risk areas [5].

In general, water safety protocols rely heavily on manual supervision by lifeguards or staff who constantly monitor the area via surveillance cameras. Although human supervision remains the primary standard, this method has significant inherent physiological and psychological limitations. Phenomena such as visual fatigue, decreased vigilance, and impaired concentration can occur even after a short monitoring period. This poses a risks of causing detection failures during critical moments [3], [6]. These conditions are exacerbated by complex environmental factors, including blinding specular reflections from sunlight, water ripple turbulence, and crowds of swimmers, which create visual disturbances that can obscure the presence of victims [7]. Consequently, in large and dynamic water bodies, it is extremely difficult for human observers to consistently maintain seamless surveillance of every blind spot [2].

This study aims to develop and evaluate a real-time automatic drowning detection system that is not only accurate but also computationally efficient, so that it can be run on resource-constrained edge devices. By validating the robustness of the YOLOv12-nano architecture against practical challenges, such as weather conditions and complex aquatic backgrounds, this study is expected to

provide an applicable, measurable solution to improve water safety standards.

II. LITERATURE REVIEW

Automation in water safety has evolved rapidly from traditional image processing methods to artificial intelligence-based solutions. Early approaches often relied on techniques such as background subtraction and optical flow. However, these methods have proven less effective in aquatic environments due to the highly dynamic, reflective nature of water surfaces [1], [4]. The advent of deep learning, particularly Convolutional Neural Networks (CNNs), has brought about a paradigm shift in the early detection of drowning victims through stronger, more adaptive feature extraction. Various studies have explored the use of surveillance cameras, both above and below the water's surface, to accelerate rescue response times [5], [6].

The You Only Look Once (YOLO) algorithm has become the de facto standard in real-time object detection because it offers an optimal balance between speed and accuracy. Previous iterations, such as YOLOv5, have been successfully applied to detect drowning victims in indoor swimming pools [8]. As the field has evolved, newer architectures like YOLOv8 have gained widespread adoption due to their improved performance in identifying swimmers under challenging visual conditions [9]. Efforts to improve accuracy have also been made through the integration of attention mechanisms to mitigate visual disturbances, such as glare [7], [10], as well as the use of augmentation techniques, such as GP-GAN, to enrich training data in rare drowning scenarios [11]. In fact, recent developments such as YOLOv10 and YOLOv11 have introduced improved end-to-end efficiency for smart monitoring applications [12], [13], [14].

Detection in aquatic environments presents unique challenges because objects are often only partially visible or distorted by water splashes. Researchers have developed multi-scale adaptive feature fusion methods, such as in the APM-YOLOv7 architecture, to improve the model's sensitivity to small objects floating on the water's surface [15]. Additionally, environmental factors such as weather changes and extreme lighting conditions require models that are not only intelligent in feature extraction but also robust under various operational conditions [2], [7]. Optimising loss functions, such as Generalised Focal Loss and Geometric Factors, has also proven crucial for improving the accuracy of bounding-box localisation for hard-to-detect objects [16], [17].

Water safety faces significant challenges due to unpredictable water dynamics and the vastness of the monitoring area [1]. Drowning incidents often occur silently and rapidly, making them prone to being overlooked without a reliable surveillance system [2], [3]. Response speed is the most critical factor; every second of delay in detecting a victim in distress significantly reduces their chances of survival [4]. Therefore, integrating smart monitoring technology capable of real-time detection is essential to strengthen public safety infrastructure in high-risk areas [5]. The most recent advancement in object detection is the YOLOv12 architecture, which introduces an attention-centric approach for real-time detection [18]. Compared to previous versions, YOLOv12 offers a significantly more efficient feature extraction mechanism by focusing on global context, which is highly relevant for recognising complex body gestures of drowning victims. The use of the YOLOv12-nano variant in this study is expected to bridge the gap between the need for a very lightweight model for edge deployment and the need for high accuracy that is robust to dynamic visual challenges in aquatic environments.

III. SYSTEM ANALYSIS AND DESIGN

The methodology applied in this study follows a systematic, structured process comprising four main phases to ensure the validity of the proposed drowning detection system. First, the data acquisition phase involves collecting high-quality datasets that accurately represent real-world aquatic scenarios, accounting for the diversity of camera viewpoints and subject poses. Second, the preprocessing phase standardizes the input data. This stage includes resizing images to a uniform resolution and normalising pixel values to achieve optimal convergence during training. Third, the model implementation phase focuses on configuring the YOLOv12-nano architecture and optimizing specific

pISSN: 2252-486X
eISSN: 2548-4710

modules, such as the backbone and heads. This optimization aims to improve detection capabilities for small objects, partially submerged subjects, and drowning incidents. Finally, the performance evaluation phase involves comprehensively testing the trained model on a validation dataset. This testing employs quantitative metrics, such as Mean Average Precision (mAP), as well as qualitative analysis via a confusion matrix, to assess the model's reliability and readiness for real-world deployment.

A. YOLOv12

The You Only Look Once (YOLO) algorithm has established itself as the leading industry-standard framework for real-time object detection. Its popularity stems from its single-stage architecture, which optimally balances detection accuracy with computational efficiency, enabling high-speed inference. The YOLOv12 architecture [18] is the latest iteration focused on enhancing feature representation while precisely managing computational load. This is achieved primarily through the implementation of the Area Attention with C2f (A2C2f) and Cross Stage Partial with C3k (C3K2) modules. The combination of these modules aims to refine feature extraction, enrich multiscale feature representations, and improve detection accuracy.

YOLOv12 offers a range of model sizes (nano, small, medium, large, and extra-large), allowing researchers to determine the optimal balance between accuracy and speed. In the context of emergency detection, such as drowning incidents, deployment on power-efficient edge devices, such as drones or surveillance cameras, is crucial. Large-scale models, such as the large and extra-large variants, are impractical for such applications because they result in excessive latency and power consumption.

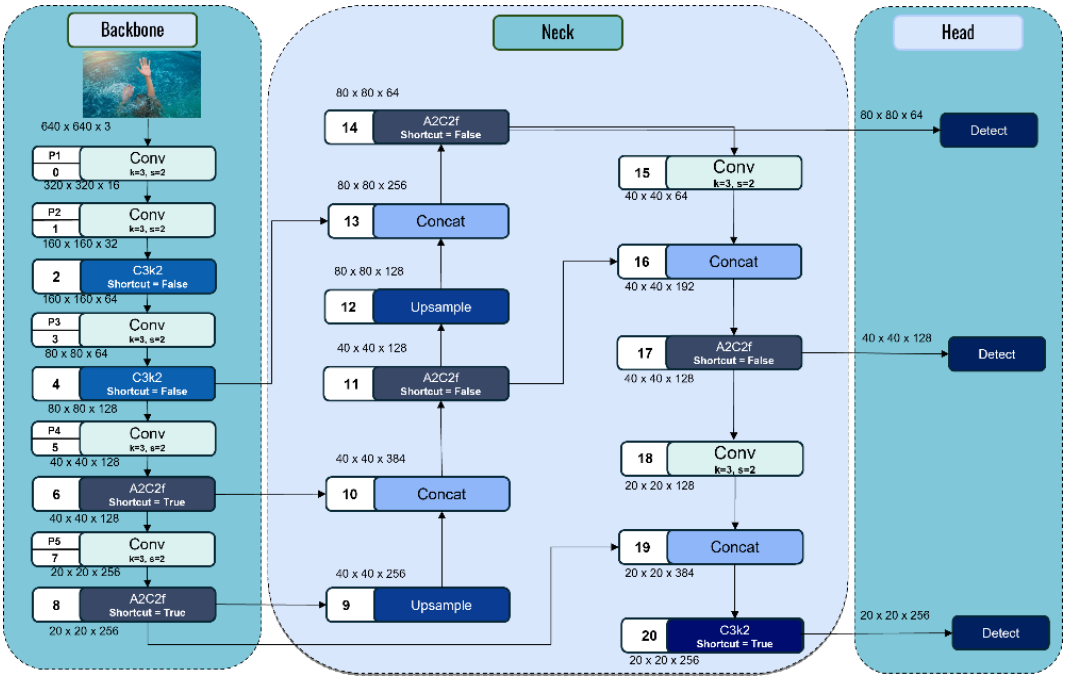

Fig. 1. The architecture of YOLOv12-nano

Therefore, this study explicitly focuses on the YOLOv12-nano variant. This variant was selected as the subject of investigation because it represents the computational capability with the minimal resource requirements. With only 2.6 million parameters and a computational load of approximately 6.5 GFLOPs, the nano variant is specifically designed to achieve maximum speed on low-power hardware. Although the small variant (9.3 million parameters, 21.4 GFLOPs) and the medium variant (20.2 million parameters, 67.5

GFLOPs) offer higher accuracy, the primary focus of this study is to evaluate the feasibility of the lightest option as a practical, real-time solution, as illustrated in Fig. 1.

1. Backbone

The backbone component in YOLOv12 serves as the primary feature extractor, processing input images to identify relevant visual patterns. This component utilizes the A2C2f and C3K2 modules to enhance feature quality and efficiency. Through its pyramidal structure, the backbone can capture features at multiple scales, enabling accurate detection of both small and large objects without significantly increasing computational load.

2. Neck

The neck component plays a crucial role in bridging the backbone and the detection head by combining multiscale feature representations. This stage integrates information from low, mid, and high-level feature maps to generate a more discriminative representation for object detection. This process involves a series of convolution operations and upsampling to align the spatial resolution across feature maps. YOLOv12 employs a combination of the Feature Pyramid Network (FPN) for top-down information propagation and the Path Aggregation Network (PAN) for bottom-up information flow. Through these hierarchical connections, the model effectively enhances contextual awareness across various layers. Additionally, the neck employs convolutional operations and a C3K2 module to improve computational efficiency and accelerate feature fusion without significantly increasing architectural complexity.

3. Head

Unlike conventional coupled architectures, YOLOv12 introduces an anchor-free detection head derived from the YOLOv8 framework. This component is designed to process objectness (object confidence), classification, and bounding box regression independently, thereby effectively improving accuracy and stabilizing the inference process. This detection head consists of two parallel branches, each with two 3×3 convolutional layers followed by a single 1×1 convolutional layer to optimize feature extraction. This module employs three detection layers with feature map resolutions of 80×80, 40×40, and 20×20, specifically tailored to detect small, medium, and large objects. For optimization, YOLOv12 uses a combination of modern loss functions. This combination includes Distribution Focal Loss (DFL) [16] and Complete Intersection over Union (CIoU) [17] for bounding-box regression, as well as Binary Cross Entropy (BCE) for classification [19].

B. A2C2f Module

The A2C2f module (Area Attention with C2f) in YOLOv12 enhances feature extraction capabilities through an optimal balance of computational efficiency and structural adaptability. In previous YOLO architectures, such as the standard C2f module in YOLOv8, feature extraction relied heavily on conventional bottleneck blocks. While effective, these standard convolutions often struggle to capture long-range dependencies and global context in highly complex visual environments. To address this limitation, the A2C2f module integrates an attention-centric paradigm directly into its cross-stage partial structure. This structural evolution is particularly beneficial for aquatic surveillance, where distinguishing a small drowning subject from dynamic, high-frequency water ripples requires a profound understanding of the global semantic context rather than merely localized textures. Operationally, this feature extraction process begins with a 1×1 convolution operation designed to reduce the input channel dimension by half (from C to 0.5C). By compressing the channel dimensions initially, the network significantly reduces the computational overhead required for subsequent deep processing, ensuring that the model remains lightweight and highly efficient for edge deployment. The output of this initial convolution is then split into two parallel paths, as illustrated in Fig. 2. One path functions as an identity connection and is passed directly as a shortcut to preserve raw gradient flow, while the other path is processed in depth using either the Area Attention (Ablock) or the C3K module, depending on the specific network configuration.

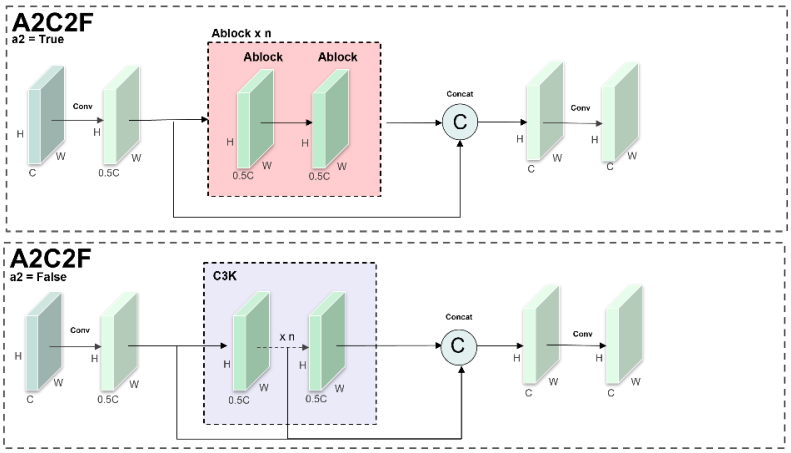

Fig. 2. The A2C2f module in YOLOv12-Nano

The Area Attention module captures global relationships through matrix multiplication between the Query (Q) and Key (K) across the entire feature space. This allows the model to focus attention on the most critical areas. Meanwhile, the C3K option is a modified version of the C3 (Cross Stage Partial) block that can adjust kernel sizes to capture spatial information with greater detail. In the final stage, the outputs from the shortcut and processing paths are concatenated, restoring the channel dimension to its original size (C). After the combination process, the features are processed by a final convolution operation to integrate and extract information comprehensively. This architectural framework ensures the model can generate far more expressive feature representations without sacrificing computational speed.

C. C3K2 Module

To address visual challenges, such as blurriness or low visibility, multiple convolutional layers are required to extract stable and discriminative features. The C3K2 module serves as an efficient feature extractor. This module acts as an enhanced version of the CSP Bottleneck, resembling the C2f structure but integrated with the design flexibility of C3K, as illustrated in Fig. 3. The process in the C3K2 module begins with a 1×1 convolution operation on the input image, which then splits the channel dimension into two equal parts (each valued at 0.5C) via a chunking operation. One channel serves as an identity connection that is passed directly to the final stage, while the other serves as the main feature extraction path.

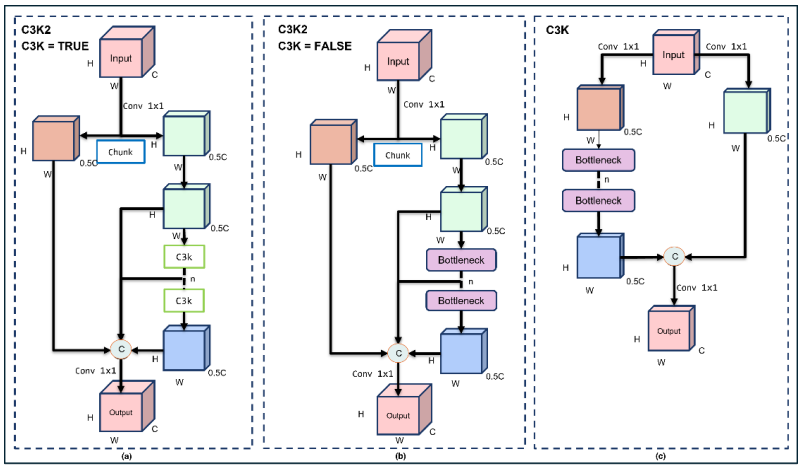

Fig. 3. The C3K2 module in YOLOv12-nano

The feature extraction pipeline supports two configurable operating modes. In the first condition, when the C3K parameter is set to True (a), features are processed using a series of C3K blocks. These C3K blocks (c) have an internal branching architecture that splits their input using two 1×1 convolutions, processes one of the paths through a bottleneck module, and then recombines them. Conversely, when the C3K parameter is set to False (b), the feature extraction path is processed using only a series of standard bottleneck blocks. This makes the overall structure of C3K2 in this mode identical to a conventional C2f module. In the final stage, the outputs from the identity branch and each layer in the feature extraction branch (from both the C3K block and the bottleneck) are recombined through a concatenation process. After this merging, the features are processed through an additional 1×1 convolution operation to enhance cross-channel interactions and restore the feature dimensions to their original size (c). This compact and adaptive design effectively produces rich and robust feature representations, making it highly optimal for handling challenging computer vision tasks.

IV. IMPLEMENTATION AND DISCUSSION

This section presents the practical implementation of the YOLOv12-nano architecture and discusses its feasibility as a real-time monitoring solution for aquatic environments. Specifically, it outlines the training configuration used to develop the model and evaluates the deployment readiness of the proposed architecture on resource-constrained devices.

A. Dataset

The dataset used in this study was sourced from a previous study conducted by Yang et al. [8]. The dataset was specifically designed for detecting drowning incidents and recorded using a DJI Mini 3 Pro drone in an indoor swimming pool environment. This dataset consists of 8,572 images with annotations for three classes of water activities: drowning, treading water, and swimming. For model training and evaluation, the data were split into a training set of 7,000 images and a validation set of 1,572 images. The original data collection [8] involved four participants simulating various postures in the water, including breaststroke, backstroke, freestyle, and drowning scenarios. This was done to ensure visual feature diversity for training a robust detection model.

B. Training and Testing Configuration

The training configuration used in this study is summarized in Table 1. The training process was conducted on the Kaggle platform using an NVIDIA Tesla P100 graphics processing unit (GPU), which provides high computational power to improve the efficiency of training and inference. The input image size was set to 640 × 640 pixels. The model was trained for 100 epochs with a batch size of 16. The optimization algorithm used was Stochastic Gradient Descent (SGD), which was selected to ensure stable convergence. The initial learning rate was set to 0.01 to control the rate of model weight updates during training. While model training was accelerated using the cloud GPU, real-world inference performance was subsequently evaluated on a Raspberry Pi 5 (quad-core ARM Cortex-A76 @ 2.4 GHz, 8GB RAM). This inference was executed entirely on the CPU to establish a baseline performance for resource-constrained edge deployments. To maximize inference speed, the trained YOLOv12-nano model was converted into the OpenVINO framework. This step ensures the model runs efficiently on the CPU.

TABLE I
TRAINING AND TESTING SETUP

Parameters	Setup
Training Platform	Kaggle
GPU	P100
Testing Platform	Raspberry Pi 5 (CPU ARM Cortex-A76)
Image Size	640 x 640 pixels
Epochs	100
Batch Size	16
Optimizer	Stochastic Gradient Descent (SGD)
Learning Rate	0.01

C. Discussion on Deployment Feasibility

Experimental results indicate a strategic trade-off that strongly supports this objective. Although the proposed YOLOv12-nano model [18] shows only a marginal difference in accuracy (0.984 mAP@50) compared to the

heavier YOLOv8-nano [9] model (0.988 mAP@50), it achieves this with a significantly higher computational efficiency (6.0 GFLOPs versus 8.2 GFLOPs). The ability to deliver such competitive performance, maintaining nearly equivalent accuracy within a 0.4% margin while reducing computational load by approximately 27%, represents a crucial advantage for maritime safety systems, undeniably given the severely limited availability of hardware resources in such environments.

In practical aquatic surveillance networks, computational efficiency is prioritized over marginal accuracy improvements. The model's lightweight design ensures high frame rates, minimizes thermal throttling, and saves energy to support continuous 24-hour operation. These endurance factors are far more critical for continuous safety monitoring than fractional differences in accuracy [1], [2].

The implications of the reduction in computational cost from 8.2 GFLOPs to 6.0 GFLOPs extend beyond processing speed; this fundamentally affects the operational viability of edge-computing-based smart camera systems [6]. A model running continuously at an 8.2 GFLOPs load will generate excessive heat, increasing the risk of device damage, a critical issue for surveillance camera units often placed in outdoor environments without active cooling systems. Additionally, thermal throttling caused by intensive processing can trigger a drop in frame rate during critical detection moments. By using the more streamlined YOLOv12-nano, the system can maintain stable, high frame-rate performance over longer periods, thereby providing continuous surveillance reliability. Therefore, the marginal 0.4% drop in mAP is a negligible trade-off compared to the substantial benefits in energy efficiency and hardware stability.

A comparison with previous research further highlights the specific contributions of this study. Although previous studies prioritized maximizing accuracy with heavier models [10], [12], they often overlooked practical implementation constraints. By demonstrating that the Nano variant can achieve competitive accuracy comparable to newer, heavier iterations such as YOLOv8 [9] and YOLOv10 [14] while using significantly fewer resources, this study provides a blueprint for the development of future efficient Edge devices [20]. The balance achieved here ensures that automated surveillance systems can become more ubiquitous, resource-efficient, and practical, ultimately contributing directly to global efforts to reduce drowning deaths.

V. TESTING

This section explains the testing and evaluation of the YOLOv12-nano model's performance. The evaluation is initially conducted quantitatively using standard metrics and is further validated through an in-depth analysis of anomalies via a confusion matrix. Additionally, a qualitative analysis of the detection results is presented to provide deeper insights into the model's robustness across various real-world scenarios.

A. Evaluation on Dataset

In this study, we evaluate the performance of the YOLOv12-nano model [18] in detecting drowning incidents using a dataset sourced from Yang et al. [8]. This dataset, designed to reflect real-world aquatic conditions, presents significant visual challenges, including variations in camera distance, diverse viewpoints, and dynamic lighting conditions. Quantitative results from a series of experiments demonstrate that the proposed YOLOv12-nano model successfully achieves an optimal balance between detection precision and computational efficiency.

As detailed in Table 2, the YOLOv12-nano model achieves a Mean Average Precision at IoU 0.5 (mAP@50) of 0.984 and a mAP@50-95 of 0.732. These metrics indicate that the model is highly capable of accurately identifying and localizing swimmers and drowning victims, with a high overlap between the predicted bounding boxes and the ground truth. A high mAP@50 score indicates that the model rarely misses targets and maintains high confidence in its detection accuracy. This is a crucial prerequisite for safety-critical applications, as detection failures can be fatal.

Beyond these numerical metrics, the architectural advantages of YOLOv12-nano are crucial to its performance in aquatic environments. The primary challenge in drowning detection is distinguishing human subjects from highly dynamic and noisy backgrounds of ripples and water splashes.

Standard convolutional layers often struggle to handle this type of high-frequency noise and tend to treat water wave patterns as potential texture features. However, integrating Area Attention with the C2f module (A2C2f) in YOLOv12 [18] effectively addresses this issue by enabling the network to focus on global semantic context rather than local texture noise. By aggregating features across a broader receptive field, the model can distinguish between random, chaotic water splash patterns and the structured albeit irregular movements of a drowning person. This architectural characteristic explains why the model maintains high precision (0.984) despite having fewer parameters than YOLOv8-nano; the model does not merely learn more features, but learns features that are far more relevant by efficiently suppressing background interference.

To put these results into context, we benchmarked the proposed model against state-of-the-art lightweight object detectors, specifically YOLOv5-nano, YOLOv8-nano, YOLOv10-nano, and YOLOv11-nano. This comparison reveals complex trade-offs between accuracy and resource consumption. YOLOv11-nano, as a very recent iteration, achieves a mAP@50 of 0.985 with 6.4 GFLOPs [13]. Similarly, YOLOv10-nano records a mAP@50 of 0.986, but requires a higher computational load of 8.4 GFLOPs [14]. By comparison, the YOLOv12-nano model we selected operates with only 6.0 GFLOPs and 2.52 million parameters, while delivering a comparable accuracy level of 0.984. This reduction in computational complexity is quite substantial.

TABLE II
COMPARISON OF YOLO VARIANTS SHOWING DETECTION ACCURACY AND EFFICIENCY

Model	GFLOPs	Parameters	mAP@50	mAP@50:95	FPS on Raspberry Pi 5
YOLOv5-nano	7.2	2,509,049	0.987	0.732	7.31
YOLOv8-nano	8.2	3,011,433	0.988	0.735	6.94
YOLOv10-nano	8.4	2,708,210	0.986	0.730	6.90
YOLOv11-nano	6.4	2,590,425	0.985	0.730	7.00
YOLOv12-nano	**6.0**	**2,520,249**	**0.984**	**0.732**	**5.21**

Specifically, YOLOv12-nano requires approximately 28% fewer GFLOPs than YOLOv10-nano [14] and consumes fewer resources than YOLOv11-nano [13], while maintaining nearly equivalent accuracy (within a 0.2% margin of difference). This efficiency benefits the model when applied in real-world scenarios. Consumer-grade surveillance cameras and drones often operate with limited power budgets and constrained processing capabilities. Models that require fewer GFLOPs enable longer flight times and lower operating temperatures, without compromising the reliability of the sinking detection system [1]. Thus, the empirical data strongly support selecting YOLOv12-nano as the optimal backbone for real-time, edge-based drowning detection systems, outperforming its predecessors in efficiency while maintaining state-of-the-art precision. Ultimately, this structural lightweightness paves the way for a more scalable and highly efficient deployment of autonomous safety infrastructure across extensive public aquatic areas.

According to Table 2, YOLOv12-nano achieves 5.21 FPS on the Raspberry Pi 5. It lags behind pure convolutional models because its branched architecture increases CPU memory access costs. Nevertheless, this model remains superior in terms of structural efficiency, with the lowest computational load (6.0 GFLOPs) and the fewest parameters. Although YOLOv12-nano is slower than other lightweight YOLO versions, it uses fewer parameters and requires less computation, which is a key advantage. This reduction in speed is a worthwhile trade-off, since 5.21 FPS is sufficient to respond to a drowning person's gestures in real time.

B. Error Analysis and Confusion Matrix

To identify areas requiring further improvement, we conducted a granular error analysis using the confusion matrix shown in

Fig. 4. The confusion matrix provides a detailed comparison between the model's predictions and the actual ground truth, revealing the distributions of True Positives (TP), False Positives (FP), and False Negatives (FN). The model demonstrated highly impressive performance by correctly identifying 2,260 objects (True Positives). Nevertheless, the matrix still revealed minor error modes worth noting. Specifically, there were only 83 cases where the background was incorrectly predicted as a subject (False Positives). These misclassifications indicate that certain dynamic elements within the aquatic background occasionally mimic the visual features of a human subject.

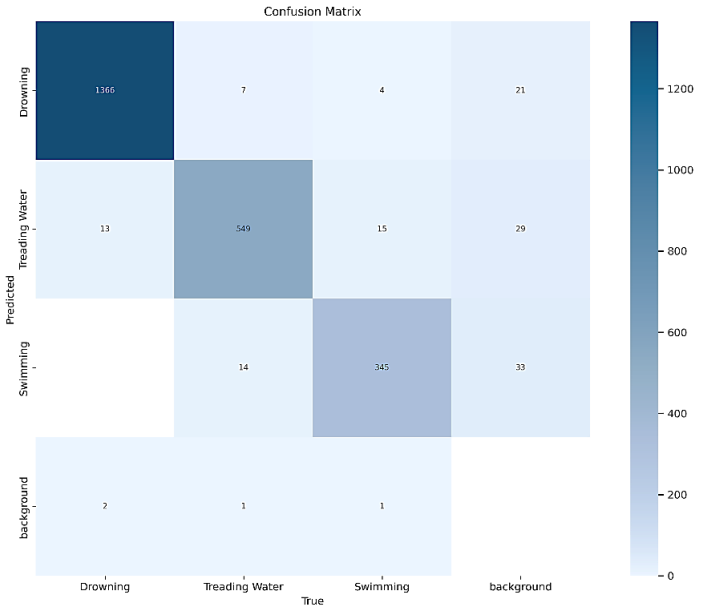

Fig. 4. Confusion Matrix of The YOLOv12-nano Model on The Selfmade Dataset

In a swimming pool environment, these false positives are most likely caused by reflections from ceiling lights on the water's surface or by complex water wave patterns resulting from the movements of other swimmers. Although the number is very low and generally less critical than false negatives in safety systems, minimizing the false-positive rate remains important to prevent alarm fatigue among human operators [2].

From a safety perspective, the most critical finding is that the model misclassified only 4 actual objects as background (False Negatives). This extremely low false negative rate demonstrates the model's reliability, given that a system's failure to detect a drowning victim is the most dangerous type of error [1]. Regarding these 4 detection failures, severe occlusion is the primary suspected cause; if a drowning victim is too deep underwater with only a minimal part of their body remaining on the surface, the available visual features become insufficient for the model to trigger a detection. The compact architecture of YOLOv12-nano [18] may inherently have limited capacity to handle these few extreme cases with very little visual information.

Furthermore, an analysis of inter-class confusion revealed 13 cases of drowning that were predicted as wading, and 7 cases of wading that were predicted as drowning. This highlights the presence of visual subtleties that pose a unique challenge for the model. Both activities involve a vertical body orientation with the head above the water surface, creating a similar visual signature in terms of the bounding box aspect ratio. The primary distinguishing feature lies in the intensity of limb movement; drowning incidents are characterized by high-frequency, irregular water splashes, whereas treading water is relatively more rhythmic [3].

Classification errors in both classes indicate that, in certain frames, motion blur from water

splashes likely obscured the positions of body parts, rendering the static image features ambiguous. As a mitigation step for future iterations, incorporating temporal information, such as analyzing the sequence of video frames rather than just a single image, could enable the system to recognize the pattern of chaos characteristic of drowning victims [6]. This approach is expected to address the ambiguities that are often difficult to categorize using static feature extractors. By transitioning from purely spatial recognition to spatio-temporal behavioral tracking, future architectures can definitively eliminate these edge-case errors without compromising the real-time processing speeds required for emergency response.

C. Qualitative Analysis of Detection Results

In addition to evaluations based on aggregate metrics, a qualitative analysis of the detection results provides deeper insights into the model's robustness across various scenarios. Fig. 5 visualizes the detection outputs on several sample frames from the dataset. These results confirm that the model can accurately classify and localize objects across the three predefined classes: drowning, swimming, and treading water.

In incidents depicting a "Drowning" condition, the model demonstrates a high level of confidence, generally ranging from 0.8 to 0.9. This is highly promising given the ambiguous nature of a drowning person's posture, which often resembles erratic swimming movements. For example, in a frame showing the subject's head nearly level with the water's surface and their arms flailing, a classic danger sign, the model accurately draws a bounding box labeled "Drowning". This indicates that the model has successfully learned specific visual features associated with the danger condition, likely by using pose-based cues captured in each frame's spatial features. Additionally, the model also demonstrates its ability to detect multiple subjects simultaneously within a single frame without experiencing a significant drop in confidence scores.

Detection performance remains robust even when the subject is recorded from varying distances. In drone-captured recordings where the subject appears much smaller (simulating aerial surveillance), the model remains capable of detecting it effectively. This robustness to scale variations is likely influenced by the multi-scale feature fusion in the YOLOv12 neck component [18], which serves to preserve semantic information. This visualization demonstrates the model's proficiency in identifying human subjects on the water's surface with high confidence across various real-world scenarios. This performance aligns with the reliability expected from lightweight detection algorithms for water rescue missions, thereby confirming its suitability for direct implementation in operational tasks.

Fig. 5. Detection Results of YOLOv12-nano on The Selfmade Dataset

VI. CONCLUSION

This study has successfully evaluated an automated drowning detection system that utilizes the lightweight YOLOv12-nano architecture. This research was motivated by the urgent need to improve water safety through automated monitoring methods. Quantitatively, the YOLOv12-nano model achieves a mAP@50 of 0.984 and a mAP@50-95 of 0.732, demonstrating high precision in detecting and classifying drowning, swimming, and splashing incidents. The most significant achievement is that this model achieves these results with optimal computational efficiency, requiring only 6.0 GFLOPs and 2.52 million parameters. Compared to other state-of-the-art variants such as YOLOv8-nano and YOLOv10-nano, YOLOv12-nano offers the most ideal trade-off; it provides comparable accuracy while significantly reducing computational load (requiring approximately 28% fewer GFLOPs than YOLOv10-nano).

These findings confirm that high-accuracy drowning detection systems are well-suited for implementation on surveillance camera networks. They enable scalable, real-time monitoring solutions. Hardware testing on a Raspberry Pi 5 successfully validated this capability, yielding a functionally sufficient processing speed of 5.21 FPS. YOLOv12-nano focuses on using fewer weights and lower computational cost than other lightweight YOLO versions. Integrating such a system can significantly support lifeguards by providing an always-on, tireless observer to detect distress signals in real time, thereby accelerating emergency response times and potentially saving lives.

Despite these promising results, implementing this automated system in real-world settings presents specific challenges that will shape future developments. A primary focus for future research is addressing the inherent ambiguity in static image detection, as early-stage drowning can visually mimic active swimming. To reduce the false negative rate observed in the confusion matrix, future iterations should incorporate temporal information such as video-based analysis using Temporal Attention mechanisms to capture the chaotic movement patterns characteristic of emergencies over time. Furthermore, to bridge the domain gap between controlled indoor pools and unpredictable open-water environments (which introduce water turbidity, strong currents, and dynamic sun glare), expanding the dataset and integrating additional attention modules will be crucial. Finally, rigorous field testing is planned to evaluate the system's robustness against practical hardware constraints, such as thermal management and power allocation in standalone surveillance units, ensuring the prototype evolves into a fully reliable safety surveillance solution.

REFERENCES

[1] S. Jalalifar *et al.*, "Enhancing Water Safety: Exploring Recent Technological Approaches for Drowning Detection," Jan. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/s24020331.

[2] M. Shatnawi, F. Albreiki, A. Alkhoori, M. Alhebshi, and A. Shatnawi, "Advances and Challenges in Automated Drowning Detection and Prevention Systems," Nov. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/info15110721.

[3] M. Shatnawi, F. Albreiki, A. Alkhoori, and M. Alhebshi, "Deep Learning and Vision-Based Early Drowning Detection," *Information (Switzerland)*, vol. 14, no. 1, Jan. 2023, doi: 10.3390/info14010052.

[4] N. Alharbi, "Exploring Advance Approaches for Drowning Detection: A Review," *Engineering, Technology and Applied Science Research*, vol. 14, no. 4, pp. 16032–16039, Aug. 2024, doi: 10.48084/etasr.7804.

[5] Q. He, Z. Mei, H. Zhang, and X. Xu, "Automatic Real-Time Detection of Infant Drowning Using YOLOv5 and Faster R-CNN Models Based on Video Surveillance," *Journal of Social Computing*, vol. 4, no. 1, pp. 62–73, Mar. 2023, doi: 10.23919/JSC.2023.0006.

[6] T. Liu, X. He, L. He, and F. Yuan, "A video drowning detection device based on underwater computer vision," *IET Image Process.*, vol. 17, no. 6, pp. 1905–1918, May 2023, doi: 10.1049/ipr2.12765.

[7] H. Sun, H. Zhao, Z. Liu, G. Jiang, and J. Zhao, "WA-YOLO: Water-Aware Improvements for Maritime Small-Object Detection Under Glare and Low-Light," *J.*

Mar. Sci. Eng., vol. 14, no. 1, p. 37, Dec. 2025, doi: 10.3390/jmse14010037.

[8] R. Yang, K. Wang, and L. Yang, "An Improved YOLOv5 Algorithm for Drowning Detection in the Indoor Swimming Pool," *Applied Sciences (Switzerland)*, vol. 14, no. 1, Jan. 2024, doi: 10.3390/app14010200.

[9] N. Alharbi, L. Aljohani, A. Alqasir, T. Alahmadi, R. Alhasiri, and D. Aldajan, "Improved Automatic Drowning Detection Approach with YOLOv8," *Engineering, Technology and Applied Science Research*, vol. 14, no. 6, pp. 18070–18076, Dec. 2024, doi: 10.48084/etasr.8834.

[10] W. Zhang, L. Chen, and J. Shi, "A Pool Drowning Detection Model Based on Improved YOLO," *Sensors*, vol. 25, no. 17, Sep. 2025, doi: 10.3390/s25175552.

[11] "Under review as a Tiny Paper at ICLR 2023 DROWNING DETECTION BASED ON YOLOV8 IM-PROVED BY GP-GAN AUGMENTATION." [Online]. Available: https://www.climatechange.ai/papers/neurips2022/37

[12] D. A. Amer, N. Y. Ibrahim, I. K. Ibrahim, A. M. Mohamed, and S. A. Soliman, "Intelligent eyes on water: YOLOv11-based real-time drowning detection system," *Journal of Supercomputing*, vol. 81, no. 12, Aug. 2025, doi: 10.1007/s11227-025-07732-7.

[13] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," Oct. 2024, [Online]. Available: http://arxiv.org/abs/2410.17725

[14] A. Wang *et al.*, "YOLOv10: Real-Time End-to-End Object Detection," Oct. 2024, [Online]. Available: http://arxiv.org/abs/2405.14458

[15] Z. Jiang, B. Wu, L. Ma, H. Zhang, and J. Lian, "APM-YOLOv7 for Small-Target Water-Floating Garbage Detection Based on Multi-Scale Feature Adaptive Weighted Fusion," *Sensors*, vol. 24, no. 1, Jan. 2024, doi: 10.3390/s24010050.

[16] X. Li *et al.*, "Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection," Jun. 2020, [Online]. Available: http://arxiv.org/abs/2006.04388

[17] Z. Zheng *et al.*, "Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation," Jul. 2021, [Online]. Available: http://arxiv.org/abs/2005.03572

[18] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-Centric Real-Time Object Detectors," Feb. 2025, [Online]. Available: http://arxiv.org/abs/2502.12524

[19]. Usha Ruby Dr.A, "Binary cross entropy with deep learning technique for Image classification," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 4, pp. 5393–5397, Aug. 2020, doi: 10.30534/ijatcse/2020/175942020.

[20] Q. Song, B. Yao, Y. Xue, and S. Ji, "MS-YOLO: A Lightweight and High-Precision YOLO Model for Drowning Detection," *Sensors*, vol. 24, no. 21, Nov. 2024, doi: 10.3390/s24216955.