

Development of a Spatial-Temporal YOLO-NAS Framework for Real-Time Drowning-Risk Detection in Open-Water Recreational Areas

Rahmat Fauzi^{1,*}, Yuhandri², Agung Ramadhanu³

¹Doctor of Information Technology Student, Universitas Putra Indonesia YPTK, Padang, Indonesia

²Doctor of Information Technology Department, Universitas Putra Indonesia YPTK, Padang, Indonesia

³Master of Informatics Engineering Department, Universitas Putra Indonesia YPTK, Padang, Indonesia

(Received: July 4, 2024; Revised: August 31, 2024; Accepted: September 11, 2024; Available online: September 23, 2024)

Abstract

Although effective in swimming pools, automated drowning surveillance in open-water recreational areas remains challenging because of dynamic conditions such as water reflections, lighting variations, wave motion, and crowded scenes. This study proposes the YOLO-NAS-SO (You Only Look Once – Neural Architecture Search – Spatial Optimization) framework to improve the spatial feature representation of YOLO-NAS while maintaining real-time computational efficiency. The framework integrates a Spatial Attention Module after the backbone feature extraction stage to emphasize relevant object features and reduce background interference. For continuous safety monitoring, the proposed framework combines a Modified ByteTrack pipeline with a temporal drowning-duration estimation module. The module monitors the persistence of drowning-risk events across consecutive video frames and generates an early warning when the event duration exceeds a predefined temporal threshold. Unlike conventional ByteTrack, the proposed extension retains the original data association mechanism while adding temporal risk assessment to the tracking process. The main contribution of this study is the integration of spatial attention-based feature enhancement with temporal risk assessment for drowning surveillance in complex open-water environments. Experimental results show that YOLO-NAS-SO achieves a mAP@50–95 of 0.7075, mAP@50 of 0.9166, precision of 0.8844, recall of 0.8817, and F1-score of 0.8830, outperforming the baseline YOLO-NAS and slightly exceeding YOLOv8 while maintaining a real-time inference time of 1.3 ms. These results indicate that combining spatial feature enhancement with temporal drowning-duration monitoring improves detection robustness and supports continuous early-warning surveillance. The proposed framework therefore provides a practical approach for intelligent drowning-risk monitoring in open-water recreational environments.

Keywords: Adaptive ByteTrack, Drowning-Risk Detection, Open-Water Surveillance, Spatial Attention Module, YOLO-NAS-SO.

1. Introduction

Open-water recreational facilities have become increasingly popular for tourism and leisure activities, attracting large numbers of visitors every year. Despite their recreational benefits, these environments present substantial safety challenges because drowning incidents often occur unexpectedly and require immediate intervention. Drowning is a time sensitive safety issue in water, as the response time is a major factor in rescuing a person. Surveillance systems in these recreational water areas should not only be able to accurately detect potential drowning events, but also aid in quick decision making for early intervention. Open water recreational areas are uncontrolled visual environments, different from traditional swimming pools, where reflections in the water, changes in lightness and darkness, wave motion, changes in viewpoint with the camera and obstruction from visitors can greatly diminish detection reliability. Recent advances in Artificial Intelligence (AI) and Internet of Things (IoT) technologies have enabled intelligent surveillance systems capable of supporting real-time monitoring and automated decision-making in safety-critical applications [1]. In addition, recent regulatory studies have emphasized the importance of strengthening safety management and monitoring strategies to reduce drowning risks in public swimming and recreational water facilities [2]. Machine learning-based surveillance systems have further demonstrated their ability to assist human operators in detecting hazardous situations through continuous video analysis [3]. Nevertheless, achieving reliable automatic

*Corresponding author: Rahmat Fauzi (rahmatfauzi.m@gmail.com)

DOI: <https://doi.org/10.47738/jads.v7i4.1515>

This is an open access article under the CC-BY license (<https://creativecommons.org/licenses/by/4.0/>).

© Authors retain all copyrights

drowning detection in complex aquatic environments remains a challenging task because environmental disturbances frequently reduce detection accuracy and delay emergency response [4].

Recent progress in deep learning has significantly improved real-time object detection performance, particularly through the development of the YOLO family of detectors. In our previous work, integrating a Spatial Attention Module (SAM) into the YOLO-NAS architecture demonstrated improved drowning and swimming detection performance by enhancing the extraction of discriminative spatial features while suppressing irrelevant background information. Recent literature reviews have also highlighted the rapid evolution of YOLO-based object detection toward lightweight architectures, improved feature extraction, and enhanced robustness in challenging environments [5]. Similar optimization strategies have been successfully applied in convolutional neural network architectures for efficient visual recognition tasks [6]. Furthermore, combining convolutional neural networks with transformer-based architectures has shown improved detection robustness in complex ecological environments [7], while attention-enhanced YOLO models have achieved higher localization accuracy for intelligent video surveillance applications [8]. Advances in image processing techniques also continue to support the development of more reliable computer vision systems for real-world applications [9].

Attention mechanisms have become an effective approach for improving feature representation in object detection models operating under challenging visual conditions. Recent studies integrating segmentation-aware attention into YOLO architectures have demonstrated improved localization performance without sacrificing computational efficiency [10]. Lightweight detection models have also been developed to support deployment on mobile and embedded platforms while maintaining real-time performance [11]. Meanwhile, evolving safety regulations for recreational water facilities increasingly require intelligent surveillance systems capable of continuous monitoring and rapid emergency response [12]. Comprehensive reviews further indicate that future object detection research should focus not only on improving detection accuracy but also on enhancing robustness and practical applicability in real-world environments [13]. In parallel, the emergence of edge intelligence has enabled advanced deep learning models to operate efficiently on resource-constrained devices, supporting low-latency inference for intelligent surveillance applications [14], [15].

Even with these advances, there are three major drawbacks. First, current drowning detection systems mainly employ frame-level object detection that can make unreliable decisions in the presence of reflections, occlusions, or cluttered scenes. Second, existing multiple object tracking techniques tend to focus on maintaining object identity instead of the temporal continuity of dangerous situations. Third, little focus has been placed on the integration of spatial feature enhancement and temporal drowning-duration assessment into a single real-time open-water recreational environment surveillance system.

To address these limitations, this study proposes YOLO-NAS-SO (Spatially Optimized), a real-time drowning detection framework specifically designed for open-water recreational facilities. The proposed framework enhances the YOLO-NAS architecture by integrating a Spatial Attention Module to improve feature representation under complex aquatic environments. Furthermore, an Adaptive ByteTrack mechanism is developed by extending the conventional tracking process with a temporal drowning-duration estimation mechanism that continuously measures how long an individual remains in a detected drowning condition. Once the accumulated duration exceeds a predefined safety threshold, the system automatically generates an early warning to support rapid intervention. Experimental evaluation demonstrates that the proposed framework improves detection accuracy while maintaining real-time computational performance, making it suitable for intelligent surveillance systems in open-water recreational environments.

2. Literature Review

2.1. Multi-Object Tracking for Intelligent Surveillance

Multi-object tracking (MOT) has become an essential component of intelligent surveillance systems because it enables continuous monitoring of object trajectories across consecutive video frames. Recent studies have demonstrated that transformer-enhanced tracking methods improve identity preservation and tracking robustness in complex environments [16]. Similarly, customized tracking algorithms have shown improved object association under severe

occlusion conditions by maintaining trajectory consistency throughout the observation period [17]. Edge-aware object detection has also contributed to more reliable monitoring in challenging environments where object appearance frequently changes [18]. In addition, dataset diversity has been recognized as an important factor for improving model generalization and tracking robustness across different environmental conditions [19]. Transfer learning techniques further enhance model adaptability when training data are limited or collected from heterogeneous imaging devices [20]. Although video analysis has continued to advance, existing surveillance systems generally focus on object detection and trajectory tracking without incorporating temporal analysis to estimate the duration of hazardous events [21], [22]. Recent drowning detection systems have also demonstrated encouraging detection accuracy; however, continuous temporal risk assessment remains largely unexplored [23].

2.2. YOLO-Based Object Detection

The YOLO family has evolved into one of the most effective object detection frameworks for real-time applications due to its balance between detection accuracy and computational efficiency. Comprehensive reviews have highlighted continuous improvements across successive YOLO generations, particularly in feature extraction, lightweight architecture, and inference speed [24]. Recent studies have further incorporated adaptive attention mechanisms to improve object representation in visually complex environments [25]. Deep learning approaches have also demonstrated the effectiveness of convolutional feature extraction for recognizing dynamic events in video sequences [26]. Meanwhile, attention-assisted tracking frameworks have improved the integration between object detection and temporal tracking, enabling more stable monitoring under challenging visual conditions [27]. The development of advanced YOLO architectures, including DAMO-YOLO and YOLO-NAS, has further improved detection efficiency while reducing computational complexity for real-time deployment [28].

2.3. Object Tracking for Behavioral Analysis

Detecting a person in a single frame is insufficient for drowning monitoring; temporal tracking is required to analyze behavioral patterns over time. Tracking-by-detection methods, such as SORT and DeepSORT, have been widely adopted [15]. However, these methods often suffer from identity switches (ID switches) during occlusions, which are frequent in crowded recreational areas. ByteTrack was introduced to address this by utilizing low-confidence detection boxes rather than discarding them, significantly improving tracking continuity in dense crowds [16]. Nevertheless, standard ByteTrack struggles with the specific motion dynamics of drowning victims, whose movements become erratic and non-linear before submersion. Therefore, modifications to the tracking algorithm, such as integrating motion prediction or trajectory smoothing tailored to aquatic environments, are necessary to ensure robust behavioral analysis [17].

2.4. Spatial Attention Mechanisms

Spatial attention mechanisms have become an effective approach for improving feature representation by enabling deep neural networks to focus on informative image regions while suppressing irrelevant background information. Spatial attention-based optimization has been shown to improve detection accuracy without substantially increasing computational cost [29]. Similar advantages have also been reported in underwater object detection, where attention mechanisms improve robustness against illumination changes, water turbidity, and complex backgrounds [30]. Recent studies integrating YOLO-NAS with attention modules have demonstrated improved object localization performance in practical surveillance applications [31]. Furthermore, the incorporation of dropout strategies has been shown to improve model generalization while reducing overfitting during deep neural network training [32].

2.5. Research Gap

Previous studies have demonstrated significant progress in object detection, attention-based feature enhancement, and multi-object tracking. However, most existing approaches treat detection and tracking as independent processes and primarily focus on improving localization accuracy or tracking continuity. Limited attention has been given to integrating temporal analysis into the tracking pipeline to estimate drowning duration and support automatic early warning generation. To address this gap, this study proposes YOLO-NAS-SO, which combines spatial feature optimization with a Modified ByteTrack framework capable of continuously estimating drowning duration based on

persistent tracking identities. The proposed framework is designed to improve not only detection accuracy but also temporal risk assessment for intelligent safety monitoring in open-water recreational facilities.

3. Methodology

This study proposes YOLO-NAS-SO, a spatially optimized object detection framework integrated with a Modified ByteTrack algorithm for real-time drowning detection in open-water recreational facilities. The proposed methodology consists of two primary phases: model development and system implementation, as illustrated in [figure 1](#). The model development phase focuses on constructing and evaluating the proposed YOLO-NAS-SO model, while the implementation phase deploys the trained model within an IoT-based surveillance system capable of continuous monitoring and early warning generation.

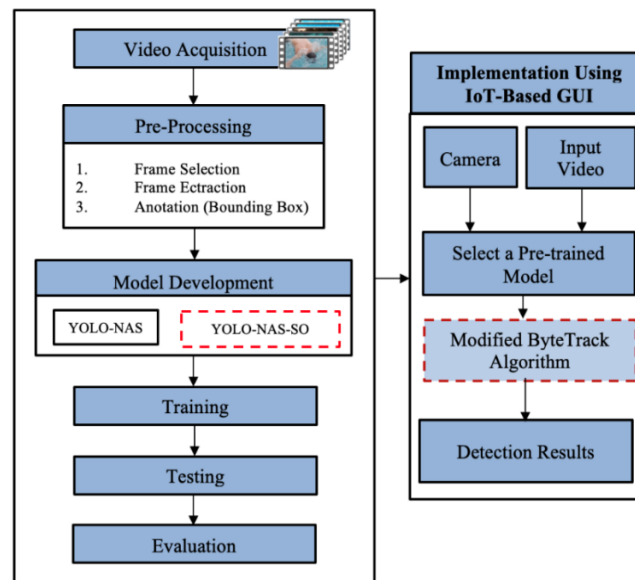


Figure 1. Proposed Research Framework

Research Framework in [figure 1](#) is composed of two phases: model development and system implementation. Video data is obtained and subsequently pre-processed during the model development phase, comprising frame selection, frame extraction and bounding-box labeling. Once the data set is prepared, the baseline YOLO-NAS model is developed and compared with the proposed YOLO-NAS-SO model. Once the model development process is complete, the model is trained, tested and evaluated to find the best performing model. The selected pre-trained model is then integrated into a graphical user interface (GUI) that runs on an internet of things (IoT) platform, which can accept input from a live camera or from uploaded video. The Modified ByteTrack algorithm is further used to process the detection output, ensuring the identity of objects, and enabling temporal monitoring of the drowning-risk event. Lastly, the system generates detection results which can help operators identify real-time potential drowning incidents.

3.1. Overall Framework

The overall framework of the proposed system is illustrated in Figure 1. The workflow begins with video acquisition from surveillance cameras installed around open-water recreational facilities. The captured videos are subsequently processed through frame extraction and manual annotation before being used for model development. Two object detection models are considered during model development: the baseline YOLO-NAS model and the proposed YOLO-NAS-SO model. The proposed architecture integrates a Spatial Attention Module (SAM) into YOLO-NAS to enhance feature representation under complex aquatic environments. Both models are trained and evaluated using identical datasets and training configurations to ensure fair comparison. After training, the best-performing model is deployed into an IoT-based graphical user interface (GUI). During real-time operation, the detection results are processed by the proposed Modified ByteTrack algorithm, which continuously monitors each detected individual and estimates the duration of drowning events. Once the drowning duration exceeds a predefined temporal threshold, the system generates an automatic warning to assist lifeguards in performing timely rescue operations.

3.2. Dataset Preparation

The dataset used in this study was constructed from two complementary sources. The primary dataset consisted of images extracted from surveillance videos recorded at open-water recreational facilities, where frames were sampled at 30 second intervals to capture representative swimming and drowning activities under real environmental conditions. The secondary dataset comprised images collected from various sources with diverse conditions, including blurred images, partially occluded objects, tilted viewing angles, and other challenging scenarios. All images in the secondary dataset were annotated using Roboflow to simplify and standardize the labeling process. After combining both datasets, a total of 4,800 annotated images were obtained for model development. The preprocessing pipeline included video acquisition, frame extraction, image selection, image annotation using Roboflow, and dataset integration, as illustrated in figure 2.

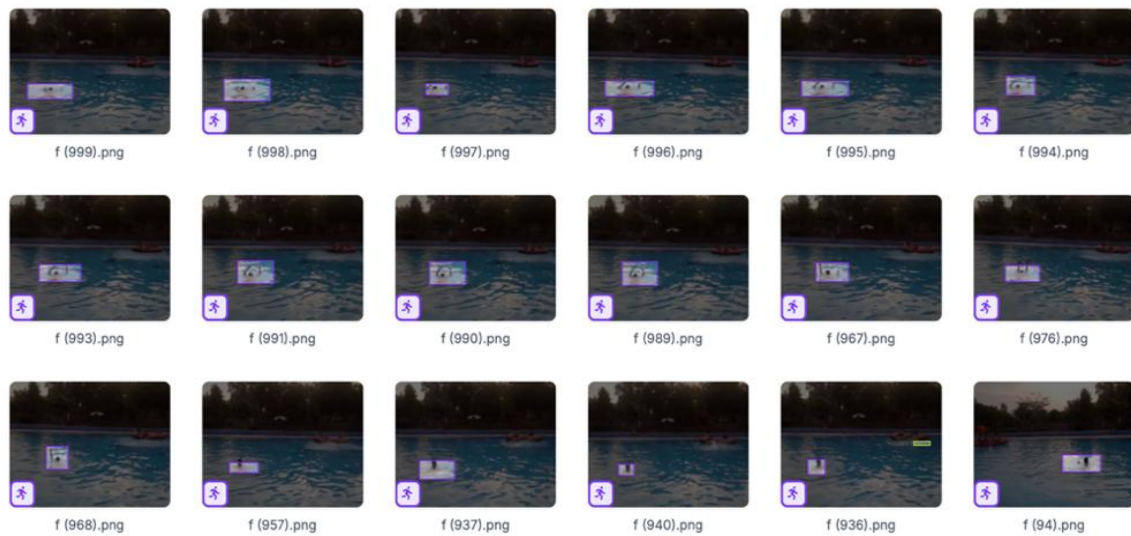


Figure 2. Examples of annotated images for the Swimming and Drowning classes.

To ensure a balanced evaluation, the complete dataset was divided into three subsets comprising 3,360 training images (70%), 960 validation images (20%), and 480 testing images (10%). The combination of primary and secondary datasets enables the proposed model to learn both domain-specific characteristics observed in open-water recreational facilities and diverse object variations obtained from large-scale public datasets, thereby improving detection robustness under different environmental conditions [19]. All images were manually annotated using bounding boxes to identify the target objects. The annotation process considered two activity classes, namely Swimming and Drowning, which were selected to support supervised object detection and facilitate accurate localization during model training. Consistent annotation across both datasets ensures reliable ground-truth information for model optimization and performance evaluation [24]. Before training, all collected data underwent a preprocessing stage to improve data quality and maintain input consistency. During frame selection, duplicated, blurred, and low-quality images were removed to reduce noise and improve dataset quality. The remaining annotated images were then resized and formatted according to the input requirements of the YOLO-NAS architecture before being used for model training. This preprocessing strategy improves data consistency and contributes to better model convergence and generalization during the learning process [32].

3.3. Proposed YOLO-NAS-SO Architecture

The proposed YOLO-NAS-SO (*You Only Look Once – Neural Architecture Search – Spatial Optimization*) architecture is developed by extending the original YOLO-NAS object detector through the integration of a Spatial Attention Module (SAM) into the feature extraction network. The objective of this enhancement is to improve the representation of discriminative spatial features while suppressing background interference commonly encountered in open-water recreational environments, such as water reflections, wave motion, illumination variation, and partial occlusions. Unlike conventional approaches that redesign the entire detection architecture, the proposed method preserves the computational efficiency of YOLO-NAS while introducing an additional attention mechanism to refine intermediate feature maps. The proposed YOLO-NAS-SO as illustrated in figure 3.

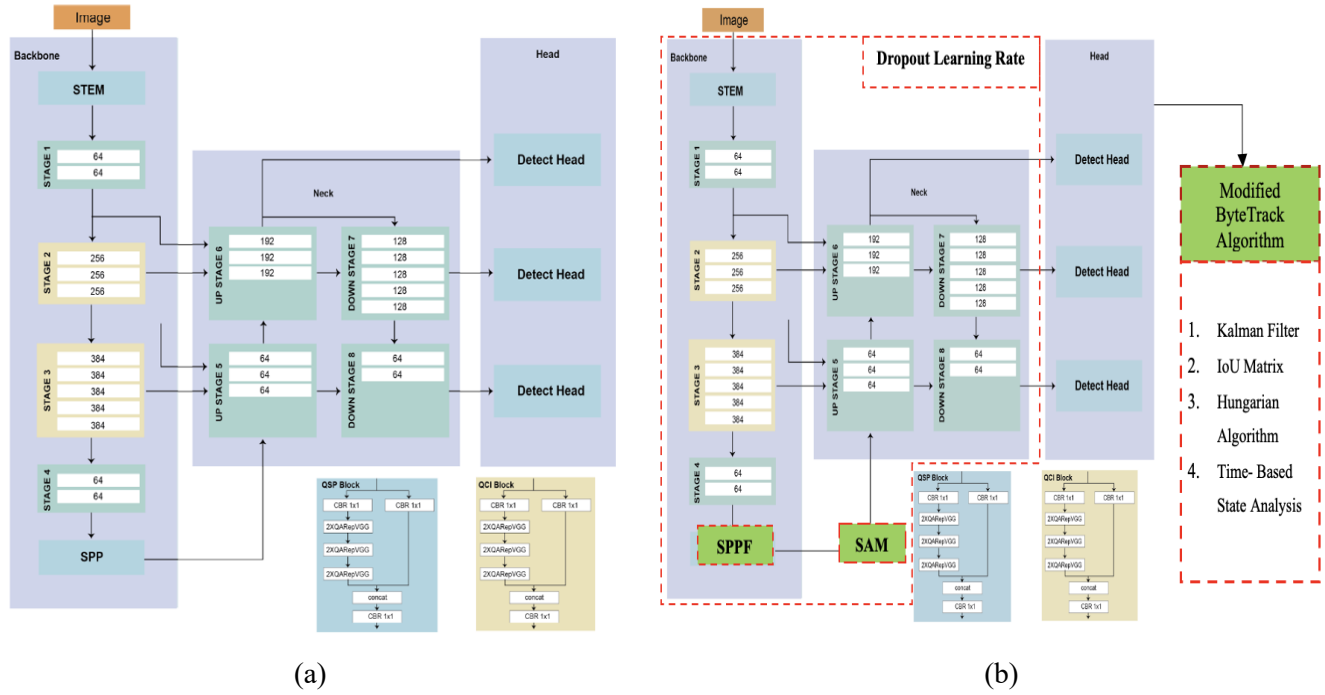


Figure 3. Comparison of the Original YOLO-NAS and the Proposed YOLO-NAS-SO Architectures (a) Original YOLO-NAS Architecture (b) Proposed YOLO-NAS-SO Architecture.

Figure 3 presents a comparison between the original YOLO-NAS architecture and the proposed YOLO-NAS-SO architecture. As illustrated in Figure 3(a), the original YOLO-NAS consists of three main components: Backbone, Neck, and Detection Head. The backbone is responsible for extracting hierarchical visual features from the input image, the neck performs multi-scale feature aggregation, and the detection head predicts object categories together with their corresponding bounding boxes. In contrast, Figure 3(b) illustrates the proposed YOLO-NAS-SO architecture. The primary modification is the insertion of a Spatial Attention Module (SAM) after the backbone feature extraction stage. The generated attention map emphasizes informative spatial regions while suppressing irrelevant background responses before the features are propagated to the neck and detection head. By refining feature representation prior to multi-scale fusion, the proposed architecture improves the localization and classification of swimming and drowning activities without altering the overall inference pipeline of YOLO-NAS.

To enhance the discriminative spatial representation of the extracted features, a Spatial Attention Module (SAM) is introduced after the Spatial Pyramid Pooling Fast (SPPF) module. The spatial attention map is generated by aggregating complementary information from average pooling and max pooling operations, followed by a convolution layer and sigmoid activation:

$$M_s = \sigma(f^{7 \times 7}([\text{AvgPool}(F_{\text{SPPF}}); \text{MaxPool}(F_{\text{SPPF}})])) \quad (1)$$

Note that the refined feature map is obtained by

$$F_{\text{att}} = M_s \odot F_{\text{sppf}} \quad (2)$$

Note:

F_{SPPF} denotes the feature map generated by the Spatial Pyramid Pooling Fast (SPPF) module

$\text{AvgPool}(\cdot)$ and $\text{MaxPool}(\cdot)$ represent channel-wise average pooling and max pooling operations, respectively

$[\cdot; \cdot]$ denotes channel-wise concatenation

$f^{7 \times 7}(\cdot)$ represents a convolution operation with a 7×7 kernel

$\sigma(\cdot)$ denotes the sigmoid activation function

$\odot$ denotes element-wise multiplication

M_s is the generated spatial attention map, and

F_{att} is the refined feature map after applying spatial attention.

The obtained attention map emphasizes informative spatial regions while suppressing irrelevant background responses. The refined feature map F_{att} is subsequently propagated to the Feature Pyramid Network (FPN) for multi-scale feature fusion before the final detection stage. Table 1 presents a comparison between the mathematical formulations of the original YOLO-NAS and the proposed YOLO-NAS-SO architectures. The modification focuses on integrating the SPPF and SAM modules to enhance feature extraction prior to the Feature Pyramid Network (FPN).

Table 1. Comparison of YOLO-NAS Formulation Before and After Development

YOLO-NAS	Proposed YOLO-NAS-SO
$F_{CSP} = CSP(F)$	$F_{CSP} = CSP(F)$
$F_{neck} = FPN(F_{CSP})$	$F_{SPPF} = SPPF(F_{CSP})$
$Y = Head(F_{neck})$	$M_s = \sigma(\text{Conv}([\text{AvgPool}; \text{MaxPool}]))$
	$F_{SAM} = M_s \odot F_{SPPF}$
	$F_{neck} = FPN(F_{SAM})$
	$Y = Head(F_{neck})$

Compared with the original YOLO-NAS formulation, the proposed YOLO-NAS-SO introduces an intermediate feature enhancement stage between the CSP block and the FPN. This modification enables the network to capture richer contextual information and refine spatial feature responses, thereby improving the discriminative capability of the detector while preserving the overall architecture of the baseline model. To clearly illustrate the architectural contribution of the proposed model, table 2 summarizes the differences between the original YOLO-NAS architecture and the proposed YOLO-NAS-SO architecture.

Table 2. Comparison between the Original YOLO-NAS and the Proposed YOLO-NAS-SO Architecture

Component	Original YOLO-NAS	Proposed YOLO-NAS-SO
Input	RGB Image	RGB Image
Backbone	Standard YOLO-NAS Backbone	Standard YOLO-NAS Backbone
Spatial Attention Module	Not Available	Integrated after Backbone
Feature Refinement	Standard Feature Extraction	Spatial Attention-based Feature Refinement
Neck	Multi-scale Feature Fusion	Multi-scale Feature Fusion
Detection Head	Standard Detection Head	Standard Detection Head
Output	Bounding Box + Class Prediction	Bounding Box + Class Prediction
Computational Complexity	Baseline	Slightly Increased
Real-time Capability	High	High
Robustness to Complex Background	Moderate	Improved

Table 2 shows the comparison of the original YOLO-NAS architecture with the proposed YOLO-NAS-SO model. The basic pipeline of both architectures is the same, which consists of RGB image input, the well-known YOLO-NAS backbone, multi-scale feature fusion in the neck, the well-known detection head, and bounding box with class prediction as the output. The main architectural difference is the introduction of the Spatial Attention Module (SAM) directly following the backbone in YOLO-NAS-SO. This extra module optimizes intermediate features by selectively highlighting the informative spatial regions and reducing background responses to information in complex aquatic environments, enhancing feature discrimination. The added overhead of incorporating SAM does not significantly add to the computation complexity, but the proposed architecture maintains the real-time capacity of the original YOLO-NAS. Moreover, the improved feature refinement further boosts the robustness of YOLO-NAS-SO for challenging open-water recreational settings, such as water reflections, illumination changes, wave motion, and partial occlusion.

3.4. Training Configuration

The proposed YOLO-NAS-SO model was trained using the AdamW optimizer with an initial learning rate of 0.001, a batch size of 16, and 100 training epochs. The learning rate was reduced compared with the baseline configuration (0.01) to achieve more stable convergence after integrating the Spatial Attention Module (SAM). A dropout rate of 0.5 was introduced to reduce overfitting and improve the model's generalization capability, particularly because the dataset combines images collected from different sources with varying visual characteristics. The batch size of 16 was selected based on the available GPU memory while maintaining stable gradient estimation during training. These

hyperparameters were determined through preliminary experiments and by considering recommendations from previous studies on YOLO-based object detection. The training configuration, as illustrated in [table 3](#).

Table 3. Training Hyperparameters Configuration of the Proposed YOLO-NAS-SO Model

Hyperparameter	Value	Justification
Optimizer	AdamW	Stable optimization and good generalization for deep CNN models
Learning rate	0.001	Provides smoother convergence after integrating SAM
Batch size	16	Selected according to GPU memory while maintaining stable training
Epoch	100	Sufficient for convergence based on preliminary experiments
Dropout	0.5	Reduces overfitting and improves generalization

The training hyperparameter setting for the proposed YOLO-NAS-SO model is shown in [table 3](#). The model was optimized by using the AdamW optimizer since it is used for stable optimization and can be used to get better generalization in deep CNN based models. The learning rate of 0.001 was chosen to achieve smoother convergence after integrating the Spatial Attention Module (SAM), and the batch size was set to 16 to accommodate the memory on the GPUs and to prevent too large a batch from causing the convergence process to get disorganized. The training process lasted for 100 epochs and it was found to be enough for convergence with preliminary experiments. Also, a drop-out rate of 0.5 was used to prevent overfitting and enhance the ability of the model to generalize. In general, these hyperparameters were chosen to be a compromise between the stability of training, the robustness of the model, and computational efficiency for real-time detection of drowning risk.

3.5. Modified ByteTrack for Temporal Drowning Monitoring

To enable continuous multi-object tracking, the proposed YOLO-NAS-SO detector is integrated with a Modified ByteTrack algorithm. This integration extends the capability of object detection from frame-based recognition to persistent object tracking across consecutive video frames, allowing each detected visitor to be assigned a unique tracking identity (Track ID) throughout the observation period [27]. The tracking information is subsequently utilized for temporal analysis, enabling the system to estimate the duration of visitor activities and identify prolonged drowning events in real time.

The workflow of the proposed Modified ByteTrack algorithm is illustrated in [figure 4](#). The tracking process consists of four sequential stages: Kalman Filtering, IoU Matrix Computation, Hungarian Matching Algorithm, and Temporal Status Analysis. Kalman Filtering predicts the object location in the subsequent frame, while the IoU Matrix Computation measures the similarity between predicted trajectories and newly detected bounding boxes. The Hungarian Matching Algorithm then assigns each detected object to its corresponding Track ID, ensuring consistent object identities throughout the video sequence.

Unlike conventional object tracking methods that focus solely on trajectory association, the proposed framework incorporates a Temporal Status Analysis module after the tracking process. This additional module continuously evaluates the activity status of each tracked visitor and accumulates the duration of the detected activity. By integrating temporal information with object tracking, the proposed system is capable of identifying prolonged drowning events and supporting automatic early warning generation.

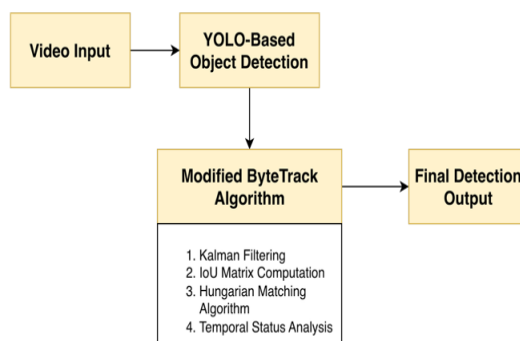


Figure 4. Workflow of the proposed Modified ByteTrack with temporal drowning-duration estimation

To emphasize the contribution of the proposed tracking framework, [table 4](#) presents a comparison between the original ByteTrack algorithm and the proposed Modified ByteTrack algorithm.

Table 4. Comparison between the Original ByteTrack and the Proposed Modified ByteTrack Algorithm

Algorithm 1. Original ByteTrack Algorithm	Algorithm 2. Proposed Modified ByteTrack Algorithm
1. Initialize the trained YOLO-NAS model 2. Initialize the ByteTrack tracker 3. For each video frame F_t do 4. Acquire the input frame 5. Perform object detection using YOLO-NAS 6. Obtain • Bounding Boxes • Confidence Scores • Class Labels 7. Separate detections into - High-confidence detections - Low-confidence detections 8. Predict object locations using Kalman Filtering 9. Compute the IoU matrix between predicted tracks and current detections 10. Perform data association using the Hungarian Matching Algorithm 11. Associate high-confidence detections with existing tracks 12. Update active tracks using low-confidence detections 13. Update Track Status • Active • Lost • Removed 14. Display • Track ID • Bounding Box • Confidence Score • Class Label 15. End For 16. If an exception occurs then 17. Display the corresponding error message 18. End	1. Initialize the trained YOLO-NAS-SO model 2. Initialize the Modified ByteTrack tracker 3. Set Duration (ID)=0 for all Track IDs 4. For each video frame F_t do 5. Acquire input frame 6. Perform object detection using YOLO-NAS-SO 7. Obtain • Bounding Boxes • Confidence Scores • Activity Labels (Swimming or Drowning) 8 Divide detections into - High-confidence detections - Low-confidence detections 9. Predict object locations using Kalman Filtering 10. Compute the IoU matrix between predicted tracks and current detections 11. Perform data association using the Hungarian Matching Algorithm 12. Update Track IDs 13. Update Track Status • Active • Lost • Removed 14. For each Active Track ID do 15. If Activity Label = Drowning then 16. Duration (ID) = Duration (ID)+Δt 17. Else 18. Duration (ID)=0 19. End If 20. Compute temporal indicator $Sk(t)=\{1, \text{ if Duration (ID)} \geq T_{th} \text{ 0, otherwise}\}$ 21. If $Sk(t)=1$ then 22. Generate Early Warning 23. End If 24. End For 25. Display • Track ID • Bounding Box • Confidence Score • Activity Label • Drowning Duration • Early Warning Status 26. End For 27. If an exception occurs then 28. Display the error message 29. End

After stable object identities have been established, the Temporal Status Analysis module performs frame-by-frame evaluation for each tracked object. At every frame, the activity status of each visitor is represented using an indicator variable. The activity state is continuously accumulated across consecutive frames and converted into real-world time based on the video frame rate (frames per second). This mechanism enables the system to measure the duration of time-dependent events, such as drowning incidents, rather than relying solely on instantaneous object detection. The temporal condition of each tracked object is represented by

$$S_k(t) \quad (3)$$

Note: S represents the condition status indicator of a visitor; k denotes the object index (tracking ID); t represents the discrete time index (frame).

Finally, the proposed YOLO-NAS-SO framework is deployed through an intuitive graphical user interface (GUI) capable of processing both recorded videos and live camera streams. The GUI visualizes detection results, tracking identities, and drowning status in real time, allowing operators to monitor visitor activities continuously and respond promptly to potential drowning incidents, can be seen in [figure 5](#).

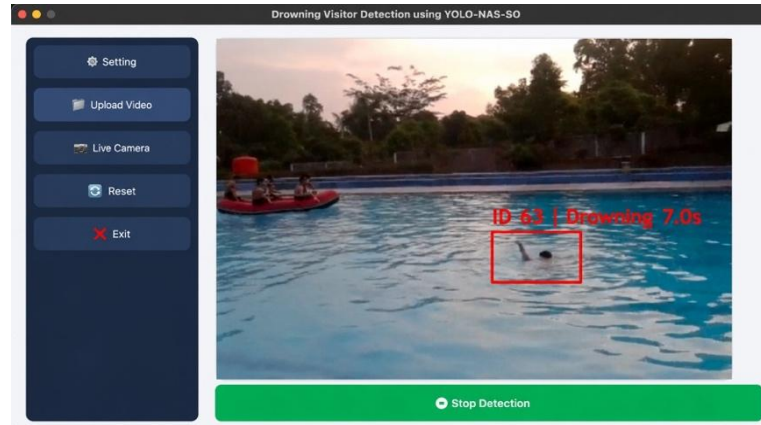


Figure 5. Real-Time Detection Result using the Proposed YOLO-NAS-SO Model

[Figure 5](#) shows the real-time detection results of the proposed YOLO-NAS-SO framework implemented through the graphical user interface (GUI). The system processes both recorded video and live camera input, displaying the detected drowning-risk object along with its tracking ID and the elapsed detection time. In this example, the system identifies a visitor in a drowning-risk condition with Track ID 63 and continuously monitors the duration of the event.

If the drowning-risk condition persists for more than 5 seconds, the system automatically triggers an early-warning alarm to notify lifeguards and nearby visitors of a potential drowning incident. This notification is intended as an early alert rather than a confirmation of drowning, allowing responders to quickly identify the victim, prepare for rescue, and provide assistance as early as possible.

The 5-second threshold was selected based on the WHO drowning safety guidance, which indicates that a victim may lose the ability to recover within approximately 30 seconds if no immediate assistance is provided. Therefore, the warning is intentionally issued well before this critical period to maximize response time and increase the likelihood of successful rescue. These results demonstrate that the proposed GUI not only provides real-time detection and tracking but also supports an effective early-warning mechanism for improving safety in open-water recreational environments.

4. Results and Discussion

4.1. Experiment Setup

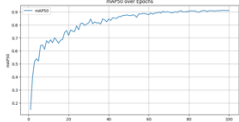
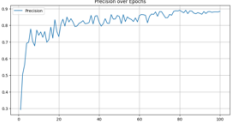
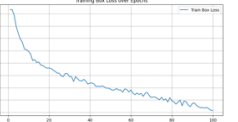
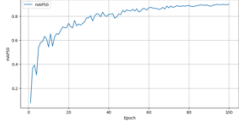
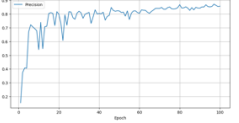
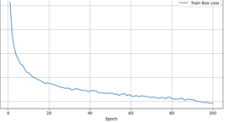
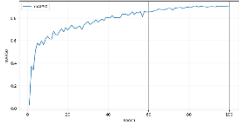
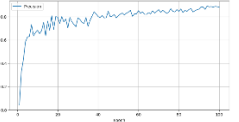
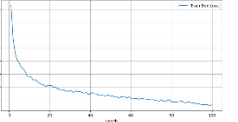
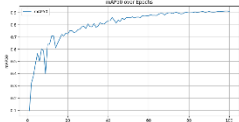
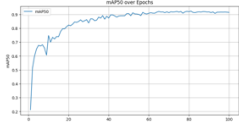
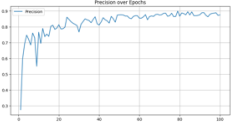
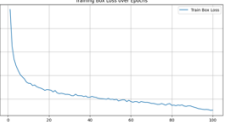
The experiments were conducted to evaluate the effectiveness of the proposed YOLO-NAS-SO framework for real-time drowning detection in open-water recreational environments. A total of 4,800 annotated images were used in the experiments, comprising 3,360 training images (70%), 960 validation images (20%), and 480 testing images (10%).

To provide a comprehensive performance assessment, the proposed model was compared with two widely adopted object detection models, namely YOLOv8 and the baseline YOLO-NAS. YOLOv8 was selected as a benchmark because it is one of the most widely used real-time object detectors, offering a strong balance between detection accuracy, computational efficiency, and deployment stability across various computer vision applications. Meanwhile, YOLO-NAS was included to evaluate the effectiveness of the proposed architectural modifications. All models were trained and evaluated using the same dataset, data partition, and experimental settings to ensure a fair and reproducible comparison. Performance was evaluated using mean Average Precision (mAP@50–95 and mAP@50), Precision, Recall, F1-score, and inference time. These metrics provide a comprehensive assessment of detection accuracy, robustness, and computational efficiency, which are essential for real-time intelligent surveillance systems.

4.2. Training Performance and Qualitative Evaluation

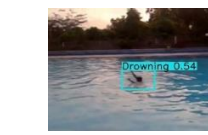
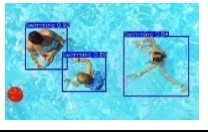
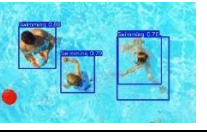
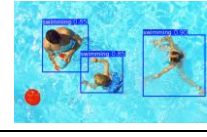
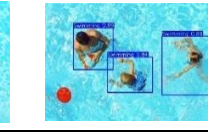
Model development and training were performed in a Python-based environment with GPU acceleration to ensure efficient model optimization. Five object detection models, namely YOLOv8, YOLO-NAS, YOLO-NAS-SPPF, YOLO-NAS-SAM and the proposed YOLO-NAS-SO, were trained using identical datasets and training configurations to provide a fair and reproducible comparison. Figure and training curves summarized in [table 5](#) illustrate the learning behavior of each model, including mAP@50, precision, and training box loss throughout the training process.

Table 5. Training Curves of the Evaluated Models

Model	mAP50 over Epochs	Precision over Epochs	Training Box Los over Epochs	Evaluation Metrics
Yolov8				EVALUATION METRICS: mAP50-95: 0.6952 mAP50: 0.9110 mAP75: 0.7002 Precision (P): 0.8838 Recall (R): 0.8636 F1 Score: 0.8736
YOLO-NAS				EVALUATION METRICS: mAP50-95: 0.6603 mAP50: 0.9003 mAP75: 0.6722 Precision (P): 0.8596 Recall (R): 0.8698 F1 Score: 0.8647
YOLO-NAS-SPPF				EVALUATION METRICS: mAP50-95: 0.6774 mAP50: 0.9051 mAP75: 0.6923 Precision (P): 0.8825 Recall (R): 0.8451 F1 Score: 0.8634
YOLO-NAS-SAM				EVALUATION METRICS: mAP50-95: 0.6939 mAP50: 0.9108 mAP75: 0.7039 Precision (P): 0.8734 Recall (R): 0.8577 F1 Score: 0.8655
YOLO-NAS-SO				EVALUATION METRICS: mAP50-95: 0.7075 mAP50: 0.9166 mAP75: 0.7269 Precision (P): 0.8844 Recall (R): 0.8817 F1 Score: 0.8830

A quantitative comparison of the evaluated models is given in [table 5](#). The results show that the proposed architectural modifications gradually improve the detection performance, from the baseline YOLOv8 and YOLO-NAS models to the integration of SPPF and SAM. Among the evaluated models, YOLO-NAS-SO achieved the best overall performance, with mAP@50–95 of 0.7075, mAP@50 of 0.9166, precision of 0.8844, recall of 0.8817, and F1-score of 0.8830. These results indicate that the integration of SPPF and SAM improves feature representation and detection robustness. The YOLO-NAS-SO model also slightly outperformed YOLOv8, confirming the effectiveness of the proposed modifications. The qualitative detection results comparison as illustrated in [table 6](#).

Table 6. Qualitative Detection Results Comparison

Video Input	YOLOv8	YOLO-NAS	YOLO-NAS-SPPF	YOLO-NAS-SAM	YOLO-NAS-SO
					
					

[Table 6](#) presents a qualitative comparison of detection results obtained from YOLOv8, YOLO-NAS, YOLO-NAS-SPPF, YOLO-NAS-SAM and YOLO-NAS-SO on representative video frames. Visually, the proposed model produces more accurate object localization, particularly in challenging situations involving partial occlusions, overlapping

visitors, and complex water backgrounds. The incorporation of SAM enables the detector to emphasize relevant spatial regions while suppressing background interference caused by water reflections and wave motion. Consequently, YOLO-NAS-SO generates more stable and consistent detections than the baseline models, making it more suitable for real-time drowning surveillance in open-water recreational environments.

4.3. Quantitative Comparison

A comparative analysis was performed to evaluate the effectiveness of the proposed YOLO-NAS-SO model in comparison with the baseline YOLO-NAS, YOLOv8, YOLO-NAS-SPPF, and YOLO-NAS-SAM models. This comparison is conducted not only at the architectural level but also based on performance metrics such as learning rate, dropout regularization, the Spatial Attention Module (SAM), mAP50, recall, precision, F1-score, and inference time. These parameters are presented together to show the contribution of SPPF, SAM, and dropout to the detection performance and computational efficiency. See [table 7](#) for a detailed comparison.

Table 7. Performance Comparison of Object Detection Models

Parameter	YOLOv8	YOLO-NAS	YOLO-NAS-SPPF	YOLO-NAS-SAM	YOLO-NAS-SO
Learning Rate	0.01	0.01	0.01	0.01	0.001
Dropout	–	–	–	–	0.5
SAM Module	Not Available	Not Available	Not Available	Available	Available
mAP50	0.911	0.9003	0.9051	0.9108	0.9166
Recall	0.8636	0.8698	0.8451	0.8577	0.8817
Precision	0.8838	0.8596	0.8825	0.8734	0.8844
F1-Score	0.8736	0.8647	0.8634	0.8655	0.8830
GFLOPs	8.1	6.3	8.4	28.4	28.4
Inference Time / FPS	3.8 ms	1.3 ms	0.9ms	1.4ms	1.3 ms

The results in [table 7](#) demonstrate that the proposed YOLO-NAS-SO model achieves the best overall performance across the main evaluation metrics. Specifically, the model attains the highest mAP50 of 0.9166, outperforming YOLOv8 0.9110 and the baseline YOLO-NAS 0.9003. In terms of precision and recall, YOLO-NAS-SO achieves 0.8844 and 0.8817, respectively, resulting in the highest F1-score of 0.8830 and indicating a better balance between detection accuracy and completeness. Compared with YOLO-NAS-SAM, the integration of Dropout and a lower learning rate further improves the model performance, particularly in recall and F1-score. This confirms that Dropout contributes to improved generalization and robustness. Moreover, the proposed model maintains efficient computational performance with an inference time of 1.3 ms, which is considerably faster than YOLOv8 at 3.8 ms. Although YOLO-NAS-SPPF achieves the fastest inference time of 0.9 ms, YOLO-NAS-SO provides a better balance between accuracy and speed, making it suitable for real-time applications. [Table 8](#) presents the comparative multi-object tracking performance of ByteTrack and the proposed Modified ByteTrack when integrated with YOLO-NAS and YOLO-NAS-SO, evaluated using HOTA, ID Switch, MOTA, and MOTP metrics.

Table 8. Performance Comparison of ByteTrack and Modified ByteTrack with YOLO-NAS and YOLO-NAS-SO

Metric	ByteTrack + YOLO-NAS	ByteTrack + YOLO-NAS-SO	Modified ByteTrack + YOLO-NAS	Modified ByteTrack + YOLO-NAS-SO
HOTA (%) ↑	3.95	33.93	4.57	67.09
ID Switch ↓	9	19	8	14
MOTA (%) ↑	-34.78	58.08	-22.95	59.49
MOTP (%) ↑	59.38	84.49	58.67	84.35

The results in [table 8](#) show that the combination of the proposed Modified ByteTrack and YOLO-NAS-SO provides the best overall tracking performance. This configuration achieves the highest HOTA 67.09% and MOTA 59.49% scores while reducing the number of ID Switches to 14, indicating more accurate and consistent object tracking across consecutive frames. In comparison, the original ByteTrack with YOLO-NAS-SO achieves a lower HOTA 33.93% and a higher number of ID Switches 19, although both methods produce similar MOTP values. The results also show that the proposed Modified ByteTrack consistently improves tracking performance over the original ByteTrack for both detection models. While the improvement is relatively small when using the baseline YOLO-NAS, a significant performance gain is achieved when combined with YOLO-NAS-SO. These findings indicate that the improved

detection capability of YOLO-NAS-SO, together with the proposed tracking modification, produces more stable object identities and more reliable multi-object tracking for real-time open-water monitoring.

4.4. Discussion

The experimental results show that YOLO-NAS-SO achieves better overall performance than YOLO-NAS and YOLOv8. The use of the Spatial Attention Module helps the model focus on important object regions while reducing the effects of background interference, such as water reflections, wave motion, and partial occlusions. The addition of Dropout also helps improve model generalization while maintaining real-time performance. In addition, the integration of temporal drowning-duration estimation with ByteTrack allows the system to monitor drowning-risk events continuously and provide an early warning when the risk persists. However, some limitations remain. The temporal warning mechanism still relies on frame-level classifications and uses a fixed 5-second threshold, which may be affected by temporary classification errors or differences in drowning situations. The system has also not yet been fully tested on embedded edge devices or evaluated with professional lifeguards in real rescue situations. Future research will focus on improving the temporal warning mechanism through temporal smoothing and adaptive thresholding. The model will also be tested on edge devices to evaluate its latency, memory usage, and energy consumption. In addition, field testing with professional lifeguards will be conducted to evaluate response time, usability, and decision accuracy in real-world conditions.

5. Conclusion

This study proposed YOLO-NAS-SO, a spatially optimized object detection framework for intelligent drowning surveillance in open-water recreational environments by integrating a Spatial Attention Module (SAM) into the YOLO-NAS architecture and incorporating a temporal drowning-duration assessment based on Modified ByteTrack. The proposed 5-second temporal threshold functions as an operational early-warning parameter to reduce transient frame-level misclassifications while enabling timely alerts, rather than representing a physiological definition of drowning. The model was trained on a combined dataset consisting of real open-water images and a public dataset containing diverse visual conditions, including motion blur, partial occlusion, illumination variations, and rotated object appearances, with separate evaluation conducted on the primary open-water subset to assess domain-specific performance. Experimental results demonstrated that YOLO-NAS-SO achieved superior detection performance over the baseline YOLO-NAS, with a mAP@50 of 0.9166, precision of 0.8844, recall of 0.8817, and an F1-score of 0.8830, while maintaining real-time inference capability. These results indicate that integrating spatial attention with temporal risk assessment improves the reliability of persistent drowning-risk detection and provides effective decision support for aquatic surveillance. Future work will focus on comprehensive tracking evaluation using adaptive temporal thresholding, real-world validation with lifeguards, and deployment optimization for embedded edge devices.

6. Declarations

6.1. Author Contributions

Conceptualization: R.F., Y., and A.R.; Methodology: Y.; Software: R.F.; Validation: R.F., Y., and A.R.; Formal Analysis: R.F., Y., and A.R.; Investigation: R.F.; Resources: Y.; Data Curation: Y.; Writing Original Draft Preparation: R.F., Y., and A.R.; Writing Review and Editing: Y., R.F., and A.R.; Visualization: R.F. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare no conflict of interest.

References

- [1] A. Tsouplaki, C. Fung, and C. Kalloniatis, “Enhancing IoT privacy with artificial intelligence: Recent advances and future directions,” *Internet Things*, vol. 34, no. 1, pp. 1–15, 2025, doi: 10.1016/j.iot.2025.101752.
- [2] T. Pobat *et al.*, “Pool safety regulations in Europe: Challenges towards a framework for sustainable seawater utilization in public swimming pools,” *Water*, vol. 17, no. 17, pp. 1–15, Sep. 2025, doi: 10.3390/w17172544.
- [3] H. Ali, A. Mehmood, N. Ahmed, M. Saeed, and A. Ijaz, “A lightweight real-time CCTV surveillance framework for the education sector using machine learning,” *J. Adv. Inf. Technol.*, vol. 16, no. 8, pp. 1072–1082, Aug. 2025, doi: 10.12720/jait.16.8.1072-1082.
- [4] M. Shatnawi, F. Albreiki, A. Alkhoori, M. Alhebshi, and A. Shatnawi, “Advances and challenges in automated drowning detection and prevention systems,” *Information*, vol. 15, no. 11, pp. 1–15, Nov. 2024, doi: 10.3390/info15110721.
- [5] H. Yu, J. Liu, and M. Lin, “A comprehensive literature review on YOLO-based small object detection: Methods, challenges, and future trends,” *Comput. Mater. Continua*, vol. 87, no. 1, pp. 1–20, 2026, doi: 10.32604/cmc.2025.074191.
- [6] M. H. Yuhandri, Yuhandri, and Sumijan, “Modified convolutional neural networks (CNN) with MobileNet-v2-186 architecture in crescent moon image detection,” *Int. Conf. Informat., Multimedia, Cyber Inf. Syst.*, vol. 2025, no. Nov., pp. 32–37, Nov. 2025, doi: 10.1109/ICIMCIS68501.2025.11326934.
- [7] A. Raza, F. Hanif, and H. A. Mohammed, “Analyzing the enhancement of CNN-YOLO and transformer based architectures for real-time animal detection in complex ecological environments,” *Sci. Rep.*, vol. 15, no. 1, pp. 1–15, Dec. 2025, doi: 10.1038/s41598-025-26645-2.
- [8] D. Nimma *et al.*, “Object detection in real-time video surveillance using attention based transformer-YOLOv8 model,” *Alexandria Eng. J.*, vol. 118, no. 1, pp. 482–495, 2025, doi: 10.1016/j.aej.2025.01.032.
- [9] A. Ramadhanu, J. Na’am, G. W. Nurcahyo, and Yuhandri, “Implementation of the affine segmentation point method and image blending techniques in creating new songket motifs,” *Int. Conf. Elect. Eng., Comput. Sci. Inform.*, vol. 2022, no. Oct., pp. 233–238, Oct. 2022, doi: 10.23919/EECSI56542.2022.9946616.
- [10] X. M. Li, X. H. Pu, W. C. Ling, and X. D. Song, “YOLO-SAM: An end-to-end framework for efficient real-time object detection and segmentation,” *Sci. Rep.*, vol. 15, no. 1, pp. 1–15, Dec. 2025, doi: 10.1038/s41598-025-24576-6.
- [11] C. Su *et al.*, “Nav-YOLO: A lightweight and efficient object detection model for real-time indoor navigation on mobile platforms,” *ISPRS Int. J. Geo-Inf.*, vol. 14, no. 9, pp. 1–15, Sep. 2025, doi: 10.3390/ijgi14090364.
- [12] T. Pobat, C. Frasca, A. Bruno, and F. Valeriani, “Thermal water-supplied swimming pools: A scoping review of regulatory frameworks, disinfection challenges, and emerging contaminants,” *Water*, vol. 18, no. 9, pp. 1–15, May 2026, doi: 10.3390/w18091050.
- [13] Z. Cai, K. Zhou, and Z. Liao, “A systematic review of YOLO-based object detection in medical imaging: Advances, challenges, and future directions,” *Comput. Mater. Continua*, vol. 85, no. 2, pp. 2255–2303, 2025, doi: 10.32604/cmc.2025.067994.
- [14] D. Ngo, H.-C. Park, and B. Kang, “Edge intelligence: A review of deep neural network inference in resource-limited environments,” *Electronics*, vol. 14, no. 12, pp. 1–15, Jun. 2025, doi: 10.3390/electronics14122495.
- [15] S. Abba, A. M. Bizi, J.-A. Lee, S. Bakouri, and M. L. Crespo, “Real-time object detection, tracking, and monitoring framework for security surveillance systems,” *Heliyon*, vol. 10, no. 15, pp. 1–15, Aug. 2024, doi: 10.1016/j.heliyon.2024.e34922.
- [16] E. Bayraktar, “ReTrackVLM: Transformer-enhanced multi-object tracking with cross-modal embeddings and zero-shot re-identification integration,” *Appl. Sci.*, vol. 15, no. 4, pp. 1–15, Feb. 2025, doi: 10.3390/app15041907.

-
- [17] W. H. E. Mg *et al.*, “Customized tracking algorithm for robust cattle detection and tracking in occlusion environments,” *Sensors*, vol. 24, no. 4, pp. 1–15, Feb. 2024, doi: 10.3390/s24041181.
- [18] Z. Zhang, C. Shuai, C. Yuan, B. Li, J. Ma, and X. Shang, “ESL-YOLO: Edge-aware side-scan sonar object detection with adaptive quality assessment,” *J. Mar. Sci. Eng.*, vol. 13, no. 8, pp. 1–15, Aug. 2025, doi: 10.3390/jmse13081477.
- [19] R. Alfred, J. Leo, and S. F. Kaijage, “Optimizing dataset diversity for a robust deep-learning model in rice blast disease identification to enhance crop health assessment across diverse conditions,” *Smart Agric. Technol.*, vol. 10, no. 1, pp. 1–15, 2025, doi: 10.1016/j.atech.2024.100726.
- [20] H. Soudbakhsh, S. R. Das, B. Hassan, and M. F. Wasiq, “Transfer learning-based ethnicity recognition using arbitrary images captured through diverse imaging sensors,” *Sensors*, vol. 26, no. 3, pp. 1–15, Mar. 2026, doi: 10.3390/s26030886.
- [21] G. D. P., A. Pathania, A. -, and S. -, “Authentication of digital MP4 video recordings using file containers and metadata properties,” *Int. J. Comput. Sci. Eng.*, vol. 10, no. 2, pp. 28–38, Apr. 2021, doi: 10.21817/ijcsenet/2021/v10i2/211002004.
- [22] P. Gong and X. Luo, “A survey of video action recognition based on deep learning,” *Knowl.-Based Syst.*, vol. 320, no. 1, pp. 1–20, 2025, doi: 10.1016/j.knosys.2025.113594.
- [23] 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,” *J. Supercomput.*, vol. 81, no. 12, pp. 1–15, Dec. 2025, doi: 10.1007/s11227-025-07732-7.
- [24] R. Sapkota *et al.*, “YOLO advances to its genesis: A decadal and comprehensive review of the You Only Look Once (YOLO) series,” *Artif. Intell. Rev.*, vol. 58, no. 9, pp. 1–15, Sep. 2025, doi: 10.1007/s10462-025-11253-3.
- [25] Md. M. Rahman, U. F. Rahim, and E. A. Taufik, “Adaptive enhancement and dual-pooling sequential attention for lightweight underwater object detection with YOLOv10,” *arXiv*, vol. 2026, no. Mar., pp. 1–15, Mar. 2026, doi: 10.48550/arXiv.2603.03807.
- [26] X.-B. Fu, S.-L. Yue, and D.-Y. Pan, “Camera-based basketball scoring detection using convolutional neural network,” *Int. J. Autom. Comput.*, vol. 18, no. 2, pp. 266–276, Apr. 2021, doi: 10.1007/s11633-020-1259-7.
- [27] M. Hou, Y. Wu, H. Shi, and X. Mu, “A two-stage multi-object tracking algorithm with transformer and attention mechanism,” *Sci. Rep.*, vol. 15, no. 1, pp. 1–15, Dec. 2025, doi: 10.1038/s41598-025-16389-4.
- [28] X. Xu, Y. Jiang, W. Chen, Y. Huang, Y. Zhang, and X. Sun, “DAMO-YOLO: A report on real-time object detection design,” *arXiv*, vol. 2023, no. Apr., pp. 1–15, Apr. 2023, doi: 10.48550/arXiv.2211.15444.
- [29] H. Ahn *et al.*, “SAFP-YOLO: Enhanced object detection speed using spatial attention-based filter pruning,” *Appl. Sci.*, vol. 13, no. 20, pp. 1–15, Oct. 2023, doi: 10.3390/app132011237.
- [30] H. Zhou *et al.*, “Real-time underwater object detection technology for complex underwater environments based on deep learning,” *Ecol. Inform.*, vol. 82, no. 1, pp. 1–15, Sep. 2024, doi: 10.1016/j.ecoinf.2024.102680.
- [31] G. B. N. I. Saluky and S. Harso Supangkat, “Enhancing abandoned object detection with dual background models and YOLO-NAS,” *Int. J. Intell. Syst. Appl. Eng.*, vol. 2023, no. 11, pp. 1–10, Nov. 2023.
- [32] W. Yulita *et al.*, “Improved human image density detection with comparison of YOLOv8 depth level architecture and drop-out implementation,” *J. Soft Comput. Explor.*, vol. 6, no. 1, pp. 33–39, Mar. 2025, doi: 10.52465/joscex.v6i1.556.