Search and Detection of People in the Water Using YOLO Architectures: a Comparative Analysis from YOLOv3 to YOLOv8

Nataliya Bilous, Vladyslav Malko and Nazarii Moshenskyi

September 23, 2024

Search and detection of people in the water using YOLO architectures: A comparative analysis from YOLOv3 to YOLOv8

Nataliya Bilous[1], Vladyslav Malko[1] , Nazarii Moshenskyi[1]

1. Kharkiv National University of Radio Electronics, Ukraine,
email: nataliya.bilous@nure.ua

Abstract. The rapid development of computer vision and deep learning technologies has significantly improved the accuracy and speed of object recognition in a variety of applications, including security, surveillance, and search and rescue. One of the key challenges in this area is the detection of people in water areas, which is crucial for improving water safety and emergency response. In this research, a detailed comparative analysis of YOLOv3 to YOLOv8, is performed to evaluate their ability for effective detection people in the water. The analysis focuses on assessing the accuracy of each version's identification of people in the water, the speed of real-time image processing, the ability to adapt to different water conditions, and the required computing resources for effective operation. The purpose of the research is to perform a detailed comparative analysis of YOLO architectures, from YOLOv3 to YOLOv8, for evaluating their ability to effectively detect people in the water area. The research not only assesses current capabilities, but also suggests directions for future innovation to improve the efficiency and reliability of detecting and tracking people on water.

Keywords: YOLO, object detection, deep learning

1 Introduction

In the modern world, the speed and accuracy of object recognition using computer vision are key in a variety of applications, from automated video surveillance to security systems and search and rescue operations. One of the critical tasks in this context is recognizing people in the water, which is crucial for preventing accidents, improving water safety, and responding effectively to emergencies. The development of deep learning algorithms, in particular You Only Look Once (YOLO) architectures, has significantly advanced the ability to recognize objects quickly and accurately in real time. Since its first introduction, YOLO has gone through several iterations of enhancements, each bringing significant improvements in accuracy, speed, and generalization ability. From YOLOv3 to the most recent version, YOLOv8, each iteration offers unique innovations that have the potential to significantly improve the effectiveness of human recognition in the water. However, to achieve the best results, it is important to have a thorough understanding of the features, benefits, and limitations of each version.

2 Purpose of the research

The purpose of the research is to perform a detailed comparative analysis of the YOLO architectures, from YOLOv3 to YOLOv8, to assess their ability to effectively detect people in the water area. The analysis focuses on several key aspects: how accurately each version can identify people in the water, how fast it processes images for real-time use, its ability to adapt to different water areas, and the computing resources required to operate efficiently.

It is planned to identify what improvements have been made to each new version of the YOLO architecture and how these changes affect their ability to detect people in the water. This includes analyzing the impact of external conditions, such as illumination and obstacles on the water, on the performance of the models. The main idea is to understand which version of YOLO is best suited for this specific task, providing the optimal combination of accuracy, speed, and efficiency in different conditions. In addition, the research will help identify potential areas for further improvements in deep learning and computer vision technologies aimed at improving human safety on the water. We aim not only to assess existing capabilities, but also to consider how future innovations can improve the process of detecting and tracking people on the water, making it more efficient and reliable.

3 General statement of the problem

To address the challenge of efficiently searching for and detecting people on water, this research focuses on analyzing and comparing different versions of YOLO architectures, which are known for their ability to recognize objects quickly and accurately in images. The importance of this analysis is to determine the optimal model that can adapt to the unique conditions of the water area, where factors such as water surface glare, water turbidity, and other variables can significantly affect the identification process. The research will evaluate each version of YOLO by the criteria of recognition accuracy, processing speed, ability to generalize to different conditions, and efficiency of computing resources. This will not only help identify the most appropriate model for the task of detecting people on water, but also identify possible areas for further research and improvement in this area.

In addition to the technical analysis, the research will also include the development of recommendations for customizing and adapting the selected YOLO models to specific water environment conditions. This will include investigating the impact of external conditions on model performance and developing techniques to optimize their performance, taking into account the challenges inherent in the water area. Thus, this research aims not only to solve the actual problem of detecting people on water using YOLO architectures, but also to contribute to the development of computer vision technologies, ensuring increased safety on water.

4 Review the Literature

Alwani [1] described a methodology for recognizing actions performed by people using joint angles and skeleton in-formation. Unlike classical methods that focus on the body contour, the presented approach was based on joint angle measurements obtained from time series of skeleton data captured by depth sensors. In turn, Chen [2] proposed a system for real-time intelligent object detection designed specifically for drones, based on Field-Programmable Gate Array (FPGA) technology and integrated into the drone. The proposed system can detect objects at a speed of 8 frames per second. In [3] Gallego used unmanned aerial vehicles equipped with multispectral cameras to search for people as part of maritime rescue operations. The aligned multispectral images were used to train a convolutional neural network (CNN) aimed at body detection. The evaluation results showed that the best classification performance obtained when combining the Green, Red Edge, and NIR channels. In particular, it was found that the precise localization method is the most suitable, showing accuracy similar to the "sliding window" method, but with a spatial localization of approximately 1 meter. Debapriya Maji [4] introduced YOLO-pose, a new method for detecting joint points and estimating the 2D position of multiple people in an image without using heat maps, based on the well-known YOLO object detection system. He obtained a new quality standard on the validation and test datasets above 90%, outperforming all existing methods without the need for additional tests and methods to improve results during testing. In [5] Bilous and Kalugin developed algorithms based on computer vision technologies to determine the position of the human body. The combination of BlazePose and the weighted distance method was found to be the best suited for position recognition, providing high ac-curacy and stability in a range of challenging scenarios. As a result, the authors conclude that the weight distance meth-od showed the best results among the position comparison methods. Daniel Hernández [6] focused on the active deployment of Unmanned Aerial Vehicles (UAVs) in various contexts and areas of application. Special attention was given to equipping these devices with a high level of intelligence and autonomy to perform complex tasks. He proposed applying deep learning techniques for the semantic segmentation of images. Experimental results demonstrated the feasibility of performing high-quality image processing of UAVs in real time using solutions developed based on Deep Neural Networks (DNNs). Xin Li [7] focused on traditional approaches to human position estimation based on heat maps. He presented an end-to-end lightweight network for human position estimation that utilizes a multiscale coordinate attention mechanism based on the Yolo-Pose network. The average accuracy of the proposed network on the COCO 2017 validation dataset increased by 4.8% while reducing the number of network parameters and computations. Nowadays, the challenge is to accurately and quickly locate people stranded on the surface of the water in flood conditions to provide them with immediate assistance. In his paper, José Gomes da Silva Neto [8] presents the initial stage of a broader research focused on the use of RGB images to analyze human behavior. At this stage, the researcher created a prototype system that combines hardware and software and can detect human postures based solely on RGB data. The choice of hardware fell on the NVIDIA Jetson Nano because of its relatively high computational efficiency compared to alternatives such as Raspberry Pi and Arduino. While searching for optimal algorithms for determining human position suitable for running on the computationally limited Jetson Nano platform, such significant developments as HyperPose, TensorRT Pose Estimation, and tf-pose estimation were identified, the latter of which was chosen for use in the project. However, the results of performance tests in terms of frames per second (FPS) showed that Jetson Nano has rather poor performance when using the selected algorithm, especially in comparison with similar systems such as NVIDIA Jetson TX2 and NVIDIA Jetson Xavier. In their article, Rakova A. O., and Bilous N. V. [9] consider the importance of tracking the direction of a person's face, which is an indicator of attention. Such data can be useful in various aspects of everyday life, including human-computer interaction, teleconferencing, virtual reality, and 3D audio rendering. Furthermore, head position detection can serve as a means of comparing the exercises performed by a person with standardized exercises. Systems based on depth cameras, which are often used for this purpose, have serious drawbacks, such as reduced accuracy in direct sunlight and the need for additional equipment. Therefore, more and more attention is being paid to recognition from 2D images, which eliminates the problems associated with depth cameras and allows them to be used both indoors and outdoors. The authors propose a landmark method that reduces the set of recorded vectors to the minimum number needed to describe head movements. They also analyze and compare existing face vector detection methods in terms of their use in the proposed approach. The results showed that the landmark method can significantly reduce the set of head direction vectors describing the movement. According to the results of the research, regression-based methods provided significantly higher accuracy and independence from lighting and partial face occlusion, so they were chosen to obtain the head direction vector in the landmark method. The findings of the research confirm the suitability of the landmark method for tracking human movements and demonstrate that methods for determining the human head vector using a 2D image can compete in accuracy with RGBD-based methods, while having fewer limitations in use. Javier Smith [10] in his research focuses on the problem of human position estimation in thermal images using convolutional neural networks and Vision Transformer architectures. The author adapts eight position estimation methods developed for visible images to the thermal domain. Due to the lack of large, labeled thermal image datasets, it is necessary to use training transfer between the visible and thermal domains, as well as a database to fine-tune the networks in the thermal domain. Dhanushree M. [11] in his article considers the problem of detecting people in the flood zone - a typical natural disaster for monsoon countries, including India. It is noted that effective detection of people in flooded areas is critical for rescue, crisis and other operational services. Since weather conditions such as rain, clouds, and fog seriously complicate the detection process, the author introduces data augmentation techniques that simulate these conditions. The author also presents the HOG-based Robust Human Object Detection (HOG_based_RHOD) algorithm, which efficiently and reliably detects people in photographs of flooded areas under various weather conditions. In their article [12], Oleh Hramm and Nataliya Bilous developed a flexible and customizable solution for cell segmentation using the Hough transform for circles and the watershed algorithm. This approach allows to optimize image processing for different datasets, providing high performance even in the presence of noise and artifacts. Considerable attention is paid to parameter tuning, which allows the algorithm to be precisely adapted to specific segmentation tasks. The results show that the developed solution has an average error of 2.69% on a sample of test images, which is an important contribution to the field of biological object image processing. Xin Wu [13] proposes a novel approach to utilize lidar data in deep learning. They focus on processing low-resolution 360° images acquired from lidar sensors. The research shows that with adequate preprocessing, lidar data can be efficiently processed by existing deep learning models for object detection and semantic segmentation.

Valarmathi [14] proposes the use of the YOLOv3 (You Only Look Once) algorithm to detect people and recognize their actions in search and rescue operations during natural disasters. This method, used with drones, allows for more efficient and faster analysis of disaster zones than traditional methods. The YOLOv3 algorithm demonstrates high accuracy (94.9%) and is able to process im-ages in 0.40 milliseconds, which is far superior to existing methods. Li Tan's research [15] improves YOLOv4 for target detection in UAV imagery by using new techniques to address the challenges of target attenuation and small object detection. It introduces a receptive field block for better feature extraction, a lightweight attention mechanism for representing features at different scales, and soft non-maximum suppression to reduce ocularity misses. The paper is organized into five chapters covering its contributions, related work, methodology, experimental validation, and conclusions, significantly advancing UAV-based monitoring and detection applications. The paper by Farzaneh Dadrass Javan [16] presents a modification of the YOLOv4 algorithm for drone recognition using visual data. The research focuses on the challenges associated with the small size of drones, their confusion with birds, the presence of hidden areas, and crowded backgrounds. The authors modified the YOLOv4 architecture, in particular the number of convolutional layers, to extract semantic features more accurately. Experimental results showed an improvement in recognition accuracy and speed compared to the baseline YOLOv4 model. Hung-Cuong Nguyen [17] explores the application of 3D human pose estimation in sports, robotics, and healthcare, focusing on fast and accurate methods. The author introduces the YOLOv5-HR-TCM (YOLOv5-HRet-Temporal Convolution Model), which is based on a 2D to 3D approach for estimating human postures in three dimensions. The model is designed to optimally perform each stage of the estimation process: face detection, 2D pose estimation, and 3D pose estimation. YOLOv6 [18] , although not an official member of the YOLO series, was inspired by the original YOLO detectors. Its primary aim is to cater to industrial applications. The main distinction from YOLOv5 lies in YOLOv6's adoption of an anchor-free method, boosting detection speed by 51%. Additionally, it incorporates the EfficientRep backbone and Rep-PAN neck and separates the regression and classification branches. For smaller models, YOLOv6 employs SIoU loss, whereas larger models utilize IoU loss. Several articles used both YOLOv6 and YOLOv5 detection models trained on various datasets. Chenhao [19] proposes an experiment that compares both models for obstacle detection for visually impaired people. In the paper, they utilized seven different YOLO detection models. They made use of a custom dataset which consists of 15 classes, 7938 labels and 2205 of them are of the class “Person”. YOLOv5 reached an overall precision of

78.1%, recall of 68.2%, and mAP@0.5 of 74.2%. YOLOv6 - an overall precision of 78.5%, recall of 71.4%, and mAP@0.5 of 78.4%. When selecting the right model scientists should consider different conditions like training time, inference speed, precision, and model size. According to Horvat et al. [20] the duration of training is smaller for smaller YOLOv5 models (YOLOv5m6 and YOLOv5s) but larger model YOLOv5x trains longer than the largest YOLOv6 model (YOLOv6-M6) - 17h 20m for 300 epochs and 11h 18m for 300 epochs respectively. Although inference time results for each v6 model are much better than corresponding v5 models of the same size, the NMS stage neglects all the improvements and results in at least twice longer total processing time. YOLOv5x processes video at twice as many frames per second as the YOLOv6-M6 model with the same complexity. According to detection results, YOLOv5 is better at handling unbalanced labels but YOLOv6 is better in object localization. YOLOv6-M reached better results in identifying the class “Person” on the COCO2017 dataset (60.3% precision, 46.0% recall, and mAP@.5 - 49.1%). These results outperform all YOLOv5 models at the detection stage. Wei Yang et al. detected vulnerable road users [21] such as motorcyclists, pedestrians, and bicyclists. They picked suitable images from different public datasets (from car cameras to regular images) and labeled them. Comparing different models they proposed their solution based on YOLOv5 as the original version performs badly on small targets. The result shows that, although the original YOLOv5 reached better FPS (98 FPS) than YOLOv6 (70.6 FPS) but still has lower accuracy - mAP@.5 85.7% when YOLOv6 mAP@.5 - 96.6%. In this experiment, only YOLOv5 exceeded 90 FPS value while all other models are far lower. Also, YOLOv5 had fewer parameters and GFLOPs compared to other models. In their research, Jinghui Yan et al. [22] developed an improved underwater object detection model based on the YOLOv7 algorithm integrated with a CBAM attention mechanism and a fast spatial pyramidal pooling through stages (SPPFCSPC) module. This research solves the problems of recognizing blurry and small objects in complex underwater environments by improving the accuracy and speed of detection. The ACFP-YOLO model has shown high accuracy compared to other state-of-the-art algorithms on URPC datasets and underwater debris detection, which is confirmed by numerous experiments and analysis. This research makes an important contribution to the development of underwater object recognition technologies. The research by Yalin Zeng et al. [23] presented the YOLOv7-UAV algorithm aimed at improving object detection in drone images. The authors make improvements to YOLOv7, including optimizing the architecture and using new methods to improve small object detection. Experiments have shown a 27% increase in detection speed compared to YOLOv7, as well as improved accuracy on the VisDrone2019 and TinyPerson datasets. These results confirm the effectiveness of YOLOv7-UAV for analyzing aerial images. Chen Haiwei and Zhou Guohui in their paper [24] describe in detail an approach to improving the YOLOv8 model for identifying student behavior. They propose an innovative module, C2f_Res2block, which combines elements from Res2Net and YOLOv8, and introduce EMA and MHSA mechanisms to im-prove the accuracy of identifying student behavior in the classroom. These innovations contribute to the overall accuracy of the model, as demonstrated by the 4.2% improvement in mAP scores. This research is important for the development of effective systems for monitoring and analyzing human activity. In the paper Gang

Wang et al. [25] the authors developed a UAV-YOLOv8 model based on the YOLOv8 algorithm. They implemented an advanced loss function, an efficient attention mechanism, and a multi-level feature fusion network. These optimizations help to improve the detection of small objects and reduce detection misses, which is especially important in UAV aerial photography scenarios. The research showed that the new model has better detection accuracy compared to the baseline YOLOv8 model and other popular models. The authors developed the UAV-YOLOv8 model based on the YOLOv8 algorithm. They implemented an advanced loss function, an efficient attention mechanism, and a multi-level feature fusion network. These optimizations help to improve the detection of small objects and reduce detection misses, which is especially important in UAV aerial photography scenarios. The research showed that the new model has better detection accuracy compared to the baseline YOLOv8 model and other popular models. J. Berndt in his paper [26] researches the accuracy of using the YOLOv8 convolutional neural network for detecting people in nadir aerial images, which is important for search and rescue operations. In this research, a unique dataset of nadir images with different ground spacing distances (GSD) was created to train YOLOv8. The impact of GSD on detection accuracy is analyzed, and it is found that networks trained on images with lower GSD (higher resolution) perform better. The results emphasize the dependence of neural network performance on the GSD of the training data, which is key to achieving high detection accuracy in aerial search and rescue scenarios.

5 Review of the YOLO architectures

YOLO (You Only Look Once) architectures represent a revolutionary approach to computational vision, offering fast and efficient real-time object recognition. Since its first introduction, YOLO has gone through several significant enhancement iterations, each bringing new improvements and optimizations, making it increasingly accurate and fast. Below is an overview of the major versions of YOLO, from YOLOv3 to YOLOv8, highlighting their key features and improvements.

5.1 YOLOv3

YOLOv3 [27], introduced in 2018, is one of the key versions of the You Only Look Once (YOLO) architecture for real-time object recognition. This version made a significant step forward from its predecessors, offering enhancements that improved recognition accuracy and speed, as well as the model's ability to identify objects of different sizes. Main features of YOLOv3:

- YOLOv3 uses three different scales of output frames, which allows the model to better detect small, medium, and large objects. This feature is critical for applications where objects may appear at different distances from the camera.
- As the basis for feature extraction, YOLOv3 uses the Darknet-53 network, which is deeper and has better performance than the Darknet-19 used in YOLOv2. Darknet-53 includes 53 convolutional layers, which enables more efficient detection of complex features in images.

- YOLOv3 introduces a new approach to predicting object classes and their sizes, using logistic regression for each object class, which allows for high classification accuracy.
- Despite the increased accuracy and ability to detect objects of different sizes, YOLOv3 remains a high-performance model capable of running in real time on modern hardware.

YOLOv3 has significantly improved the ability of computer vision systems to recognize objects accurately and quickly in images or videos. However, like any deep learning model, it has its limitations, particularly in cases where objects are highly overlapping or where there are many small objects in the scene. The introduction of YOLOv3 was an important milestone in the development of object recognition algorithms, laying the groundwork for further innovations in this area. This model demonstrated that it is possible to achieve high accuracy and speed at the same time, which paved the way for the development of subsequent versions of YOLO, each of which continues to improve these key aspects.

5.2 YOLOv4

YOLOv4 [28], introduced in 2020, continues the innovative path set by previous versions of the YOLO architecture. This version focuses on achieving the highest possible accuracy and speed of object recognition on a variety of hardware, including power-limited devices. YOLOv4 introduces several innovations and optimizations that make it one of the most powerful models for computer vision applications.

Main features of YOLOv4:

- One of the key goals of YOLOv4 was to ensure high performance not only on powerful GPUs, but also on less powerful hardware. This makes YOLOv4 available for a wider range of applications, including embedded systems and mobile devices.
- YOLOv4 integrates the latest advances in deep learning and computer vision, including techniques such as mish-activation, Cross-Stage Partial networks (CSPNet), Self-Adversarial Training (SAT), and others. These innovations help improve recognition accuracy and speed.
- YOLOv4 implements an automatic process for finding optimal hyperparameters and network structures, which maximizes the model's efficiency without the need for manual tuning.
- Through the use of advanced training and optimization techniques, YOLOv4 demonstrates an improved ability to detect objects in challenging environments such as low light, overlapping objects, and high object density in the image.

YOLOv4, with its high accuracy and speed, is a significant contribution to the field of computer vision. However, like any complex system, it has challenges, particularly in cases of extremely small or difficult objects to detect. Nevertheless, YOLOv4 remains one of the most popular and widely used models for real-world object recognition tasks, setting the standard for future developments in this field.

5.3 YOLOv5

YOLOv5 [29], launched in 2020, is not an official continuation of the YOLO line from its creator Joseph Redmon, but it quickly gained popularity in the community for its affordability, high performance, and ease of use. It was developed and published as an open-source project that continues the evolution of YOLO for object recognition.

Main features of YOLOv5:

- The YOLOv5 has been optimized to run quickly on a variety of hardware, including power-limited devices, making it highly flexible for a variety of applications.
- The ease of setup and use of YOLOv5 makes it accessible even to those new to computer vision, offering ample opportunity for customization and experimentation.
- YOLOv5 offers several model options with different numbers of parameters, from lightweight to heavier versions, allowing you to choose the optimal balance of speed and accuracy depending on your project needs.
- Improved accuracy: Despite its speed, the YOLOv5 delivers high recognition accuracy, rivaling the heavier models in the other YOLO series.

YOLOv5 continues to innovate in the area of fast and accurate object recognition, while delivering significant improvements in ease of use and accessibility. However, as with any technology, its performance can vary depending on specific application conditions, such as the variety of objects, lighting conditions, and the level of object overlap. YOLOv5 has become a popular choice for developers and researchers looking for an effective solution for a wide range of object recognition tasks, from smart video surveillance to autonomous driving systems. It has set new standards for the ratio of speed, accuracy, and ease of use in computer vision.

Although YOLOv5 is known for its high performance and efficiency in real-time object recognition, it is important to note that its training process requires significant computing resources. This is due to several factors. YOLOv5, like most deep neural networks, contains millions of parameters that need to be optimized during the training process. This optimization requires a significant amount of computation, especially with large datasets that are typically used to train computer vision models. To achieve high accuracy and overall model generalization capability, YOLOv5 is trained on large and diverse datasets. Processing and analyzing such a large number of images requires significant computing power, especially to ensure real-time training efficiency. To optimize the training process of deep learning models, in particular YOLOv5, it is recommended to use specialized hardware such as graphics processing units (GPUs). GPUs significantly speed up training due to their ability to process large amounts of data in parallel. However, the cost and availability of powerful GPUs can be a barrier for some researchers and developers. YOLOv5 training requires a large amount of RAM and video memory to store intermediate data such as model weights, gradients, and image batters. This increases the overall computational resource requirements needed for effective training. Despite these challenges, YOLOv5 remains one of the most effective and widely used models for real-time object recognition, offering high accuracy and speed. Optimization of resource utilization and hardware development continue to help reduce these limitations, making YOLOv5 training and deployment increasingly affordable.

5.4 YOLOv6

YOLOv6 [18, 30] is a modern version of the famous YOLO series of architectures for real-time object detection. Developed by the Meituan team, YOLOv6 has made a number of innovative improvements to the architecture and training methodology aimed at improving accuracy and processing speed without significantly increasing computational costs.
Main features of YOLOv6:

- Bidirectional Concatenation Module (BiC). This module improves object localization by enabling the model to integrate information from different layers of the network more efficiently. It provides a performance boost with minimal speed degradation, increasing the overall accuracy of the system.
- Anchored training strategy (AAT). AAT is used to combine the benefits of anchored and unsupervised approaches, optimizing the learning process and increasing the efficiency of inference. This allows YOLOv6 to achieve high accuracy while maintaining processing speed.
- Improvements in the model's spine and neck architecture allow YOLOv6 to achieve impressive results on large datasets such as COCO, especially at high input resolutions.
- Self-distillation strategy. This strategy aims to improve the performance of smaller models by reinforcing an additional regression branch during training and removing it during output. This helps to avoid reducing the processing speed without losing accuracy.
- YOLOv6 provides flexibility in backbone selection, offering support for CSPDarknet, EfficientNet, and ResNet, among others. Users can tailor the backbone to their needs by replacing CSPDarknet53 with any other CNN architecture that is compatible with YOLOv6 input and output data sizes.

With its innovative features, YOLOv6 is being used in a wide range of applications, from video surveillance and security systems to smart image analysis and autonomous transportation systems, where high accuracy of real-time object detection is required. YOLOv6 is a significant step forward in object detection algorithms, offering high performance, accuracy and flexibility to meet the needs of modern computer vision applications.

5.5 YOLOv7

YOLOv7 [31] is one of the newest versions in the YOLO series of real-time object recognition architectures. YOLOv7 continues the tradition of innovation established by previous versions, offering significant improvements in accuracy and speed while maintaining high efficiency.
Main features of YOLOv7:

- YOLOv7 introduces a number of technological innovations that significantly improve the accuracy of object recognition, including improved image processing algorithms and optimized network architecture.
- Despite the increased accuracy, YOLOv7 maintains a high processing speed, enabling it to be used in applications that require instant response, such as video surveillance, autonomous driving, and interactive systems.
- YOLOv7 runs efficiently on a wide variety of hardware, including both high-performance GPUs and less powerful computer systems, making it available for a wide range of applications.
- The model's architecture is designed to be easily adapted to specific object recognition needs, allowing users to customize the model for optimal performance depending on the task at hand.

YOLOv7 is setting new standards in object recognition accuracy and speed, but like any advanced technology, it has its challenges. The main ones include the need for significant computing resources to train the model, especially when working with large datasets, as well as the need for deep machine learning knowledge to effectively adapt and customize the model for specific tasks. YOLOv7 continues to advance the field of computer vision by offering powerful tools for developers and researchers interested in real-time object recognition. Its high performance and adaptability make it an ideal choice for a wide range of applications, from industrial monitoring to interactive technology development.

5.6 YOLOv8

Ultralytics YOLOv8 [32] represents a significant advancement in object detection, instance segmentation, and image classification technologies. This version provides unsurpassed accuracy and high speed, allowing for real-time object detection without delays, distinguishing it from previous versions.
Main features of YOLOv8:

- YOLOv8 provides users with the ability to customize the model architecture to meet specific needs and requirements. This flexibility is key to adapting the model to a variety of use cases, from simple applications to complex image analysis systems.
- With the introduction of new adaptive training capabilities such as loss function balancing and the Adam optimizer, YOLOv8 improves training speed and delivers better accuracy and faster model convergence. These innovations contribute to the overall improved model performance.
- The model has advanced image analysis capabilities, allowing not only to detect objects, but also to recognize actions, color, texture, and establish connections between objects. This opens up new horizons for applications in the security, medical, retail, and other industries.
- New data augmentation techniques help the model better cope with image variations, such as low resolution or occlusion, improving the accuracy of real-world object detection.

  YOLOv8 offers support for multiple backbones, allowing users to choose between CSPDarknet, EfficientNet, ResNet, and others, or even customize their own CNN architecture. This provides great flexibility and the ability to adapt the model to specific tasks. YOLOv8, like any other deep learning model, has its limitations that can affect its effectiveness in certain scenarios. Potential limitations of YOLOv8:

- Detecting very small objects: Although YOLOv8 has improved its ability to detect small objects compared to previous versions, it may still have difficulty with very small or distant objects, especially in environments with a lot of noise or clutter.
- Dependence on data quality: The performance of YOLOv8 is highly dependent on the quality and variety of data used in training. Insufficiently representative or limited training data can lead to poor model generalisation and poor performance on new or unfamiliar images.
- Handling collisions and overlapping objects: In scenes with a high level of overlap or collision between objects, YOLOv8 may have difficulty accurately detecting individual objects, especially if the objects have similar characteristics or are in close clusters.
- Difficult weather conditions: The performance of YOLOv8 may be adversely affected by difficult weather conditions, such as rain, fog, or snow, which may reduce the visibility of objects in images or videos.
- Computational requirements: Despite improvements in computational efficiency, YOLOv8 still requires powerful computing resources for real-time training and inference, especially when dealing with high resolution or large amounts of input data.
- Overall adaptability: While YOLOv8 is highly adaptive to a variety of object detection tasks, there are specific scenarios or domains where it may require significant modifications or tweaks to achieve optimal performance.

Understanding these limitations is important for choosing the most appropriate architecture for a particular task and for improving the model through further research and development. YOLOv8 can be adapted to specific tasks through fine-tuning using a specialized dataset. This process allows the model to "remember" the knowledge gained during previous training and adapt it to new conditions, improving the accuracy of detecting specific objects. Fine-tuning and full training of YOLOv8 can be performed through Python code or via the command line interface, providing flexibility and accessibility to a wide range of users. These innovations and enhancements make YOLOv8 an ideal tool for developers and researchers looking for effective computer vision solutions for a wide variety of applications.

6 Comparative analysis of YOLOv8 with other neural network models

The YOLO architecture is known for its ability to quickly detect objects in images with high accuracy. The latest version, YOLOv8, has made significant improvements in speed and accuracy, making it a potentially ideal choice for applications such as searching for and detecting people on water. To evaluate the performance of YOLOv8, it is important to compare it to other advanced object detection models.

Faster R-CNN is a model that combines high accuracy with the ability to handle complex scenes efficiently. Despite its high accuracy, Faster R-CNN has a significantly slower detection rate than YOLOv8, which can be a limiting factor for real-time applications such as water body monitoring. YOLOv8 outperforms Faster R-CNN in terms of processing speed while providing comparable accuracy, making it more suitable for rapid emergency response. SSD (Single Shot Multibox Detector) is distinguished by its ability to detect objects quickly with reasonable accuracy. It uses fixed anchor frames to detect objects of different sizes at different scales, making it effective for a wide range of applications. However, YOLOv8 performs better when detecting small objects and in challenging environments, such as water environments with variable visibility and light refraction. EfficientDet is one of the newest object detection models that is optimised for maximum efficiency without significant loss in accuracy. This model uses an efficient scalable architecture and an optimised loss function to detect objects of different scales with high accuracy. Despite its optimisation for efficiency, YOLOv8 outperforms EfficientDet in terms of processing speed, which is a key factor for real-world applications where time is of the essence. In addition, YOLOv8 performs better in detecting people in complex aquatic environments, where the model's performance is determined not only by accuracy but also by its ability to quickly adapt to dynamic conditions. For a structured comparison of YOLOv8 with other advanced object detection models, you can create a table that displays the key characteristics of each architecture (Table 1).

Table 1. Structured comparison of YOLOv8 with other neural network models

Feature	**YOLOv8**	**Faster R-CNN**	**SSD**	**EfficientDet**
Speed	Very High	Low	High	High
Accuracy	High	Very High	Moderate	Very High
GPU Resources	Moderate	High	Moderate	Low
Complexity	Moderate	High	Low	Moderate
Use Case	Real-Time	Detailed Analysis	Broad Range	Optimization
Features	Fast Detection	Precise Localization	Scalability	Efficiency

YOLOv8 sets new standards for speed and accuracy in object detection in challenging environments, such as people detection on water. Its comparison with other advanced models such as Faster R-CNN, SSD, and EfficientDet highlights its advantages in processing speed and adaptability. While each of the models reviewed has its own strengths depending on the specific application, YOLOv8 stands out as a particularly powerful tool for fast and accurate real-time object detection. This makes it ideal for applications where a quick response is crucial, especially in search and rescue missions on the water.

7 Experiments

7.1 Data preparation

In the experiment to compare versions of YOLO architectures for detecting people in the water, three key datasets were used: COCO2017, SARD (Search and Rescue Dataset), and SeaDronesSee. These datasets were chosen to provide a broad coverage of potential scenarios for recognizing people in different environments, including water. The COCO2017 [33] dataset, with its extended set of 118,000 training and 5,000 validation images, was used as the basis for the overall training of the models for object recognition. Given the goal of the experiment, special attention was paid to images containing scenes with water and people in these contexts. The SARD [34] dataset, containing 1,981 manually labeled images extracted from video footage of simulated search and rescue scenarios, was used to train the models on specific conditions. The images included a variety of terrain types and human activity, allowing the models to adapt to a wide range of detection scenarios. specializes in search and rescue scenarios, including the detection of people in difficult conditions. This dataset helps to determine how well YOLO models can identify people in critical situations on the water where speed and accuracy are key. SeaDronesSee [35] includes images of marine areas captured by unmanned aerial vehicles. With 8,930 training, 1,547 validation, and 3,750 test images, SeaDronesSee facilitated model training and evaluation on a large amount of data from maritime scenarios. The drone imagery provided a unique perspective for detecting people in the water.

In general, data preparation prior to model training included thorough normalization, augmentation, and specific processing to optimize the images for the specific conditions of the experiment. This provided a comprehensive framework for training, evaluating, and comparing the performance of different versions of YOLO in the task of detecting people on water, providing the models with the ability to adapt to a variety of scenarios and conditions, from search and rescue operations to water monitoring.

7.2 Training YOLO architectures

The process of training the YOLO architectures for the task of detecting people on water was carried out considering the unique challenges posed by this specific task. Thanks to the detailed COCO2017, SARD, and SeaDronesSee datasets, a solid foundation was created for effective model training. This process involved several key steps aimed at maximizing the performance and accuracy of each version of YOLO. Before the training, the models were adapted to the specifics of the task of detecting people on water. This meant selecting and optimizing images from datasets that best represent potential scenarios that the model could encounter in the real world. Data augmentation was also applied to create a variety of conditions, such as changes in lighting, weather conditions, and other factors that affect recognition. The training of each version of YOLO was customized to ensure optimal performance. Given the wide variety of scenarios present in the datasets used, training parameters such as learning rate, batch size, and number of epochs were carefully chosen for each model. This allowed us to strike a balance between the training speed and the ability of the models to generalize learning on a variety of data. During the training process, the performance of the models was evaluated using validation datasets. This made it possible to identify potential problems with overtraining or undertraining and to adjust training parameters in a timely manner. The key metric for evaluation was average precision (AP), which provided information about the models' ability to accurately localize and identify people in the water. At this stage, detailed optimization of the architectures was performed. For training the YOLO architecture, we used the NVIDIA TESLA A-100-80GB GPU. The choice of this powerful GPU was driven by the need to provide high computational power, which is critical for efficient training of deep neural networks, especially when working with large datasets and complex architectures such as those characterizing YOLO models. NVIDIA TESLA A-100-80GB delivers exceptional performance with its Ampere architecture, which is specifically designed to accelerate deep learning operations. With 80GB of GPU memory, this processor is capable of processing huge amounts of data, allowing you to train complex models with high speed and efficiency. Using the TESLA A-100 not only significantly reduced model training time, but also allowed us to run experiments with high resolution images and complex architectures without compromising on accuracy and performance. This GPU also helped to improve the efficiency of the data augmentation process and the implementation of additional training techniques such as fin-tuning and additional hyperparameter tuning. In addition, the use of such a powerful computing platform made it possible to analyze in detail the behavior of each version of YOLO in different conditions and with different configurations, determining the optimal parameters for maximizing the performance of the models. This, in turn, contributed to a deeper understanding of the mechanisms of object detection and opened up new opportunities for further improvement of machine vision technologies. Thus, the use of NVIDIA TESLA A-100-80GB in the experiment with human detection on water using YOLO architectures has significantly improved the quality and speed of the research, demonstrating the importance of choosing a powerful computing platform for training modern deep learning models. A special optimization formula was used to adjust and fine-tune the neural network parameters, ensuring optimal performance:

$$E(\theta) = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} L(yi, f(xi;\theta)) \tag{1}$$

where $E(\theta)$ – general error on the dataset, L - loss function that determines the difference between the actual and predicted values, yi - is the actual value of the label for the i-th sample, $f(xi;\theta)$ - is the predicted value of the model for the i-th sample with parameters θ, N - number of samples in the dataset.

The training process is based on the YOLO architecture using Adam optimizer. To determine the accuracy of the model, F1-Score is used. Table 2 presents the comparative performance of different versions of YOLO architectures for object recognition after 30 epochs of training. It includes such metrics as Precision, Recall, average accuracy at a threshold of 0.5 (mAP@0.5), and F1-Score for each version of the model from YOLOv3 to YOLOv8.

Table 2. YOLO architectures performance.

Architecture	Precision	Recall	mAP@0.5	F1-Score
YOLOv3	0.67	0.68	0.675	0.675
YOLOv4	0.81	0.83	0.82	0.82
YOLOv5	0.73	0.73	0.73	0.73
YOLOv6	0.62	0.62	0.62	0.62
YOLOv7	0.68	0.68	0.68	0.68
YOLOv8	0.87	0.89	0.88	0.88

YOLOv6, performing lower on key metrics such as precision, recall, and mAP@0.5, stands out as a model that needs further tuning to improve its ability to identify true positives. The recommendation of Li et al. [18] to increase the number of training epochs from 300 to 400 as a way to improve performance and mAP is valuable because it is based on empirical data and can help in detecting more objects in images with higher accuracy. At the same time, the high memory consumption of the YOLOv5GPU during training, especially with the 64 batch size of 73.2 GB, indicates significant resource requirements. This can be a challenge for researchers and developers, especially in resource-constrained environments, and requires consideration of alternative optimization strategies or the use of more efficient neural network architectures to reduce computing resource requirements without significant performance loss.

Based on this data, after 30 epochs of training, YOLOv8 showed the highest overall performance, recording the best mean accuracy (mAP) of all YOLO versions. The results of the training showed that each version of YOLO has its own strengths and weaknesses in the context of the task of detecting people in the water. Based on this data, additional model optimizations were made to further improve their performance. This included fine-tuning the models using specific datasets and adapting the architectures to the specific challenges identified during the experiment. Thus, the training process of the YOLO architectures was deeply integrated with the unique requirements of the waterborne detection task, demonstrating the importance of careful data preparation, proper training setup, and an adaptive approach to model evaluation and optimization.

8 Results

After performing thorough experiments using different versions of the YOLO architectures for the task of recognizing people on water, significant data was obtained that reflects the performance of each model. The models were trained on a balanced dataset including images from COCO2017, SARD, and SeaDronesSee to evaluate their performance in a variety of waterborne detection environments. The analysis of the training results after 30 epochs showed that the YOLO models demonstrate different levels of performance, in particular in such key metrics as accuracy, recall, average accuracy at a threshold of 0.5 (mAP@0.5) and F1-score. The YOLO architecture classifies objects using a single neural network pass to simultaneously predict multiple classes and locations of objects in an image. The model divides the image into a grid and for each cell predicts the bounding box, the probability of an object being in the box, and the object's classification. This allows YOLO to detect and classify objects quickly and with high accuracy, making it ideal for real-time applications. Figure 1 shows the classification of people on the water using surfboards with YOLOv7.

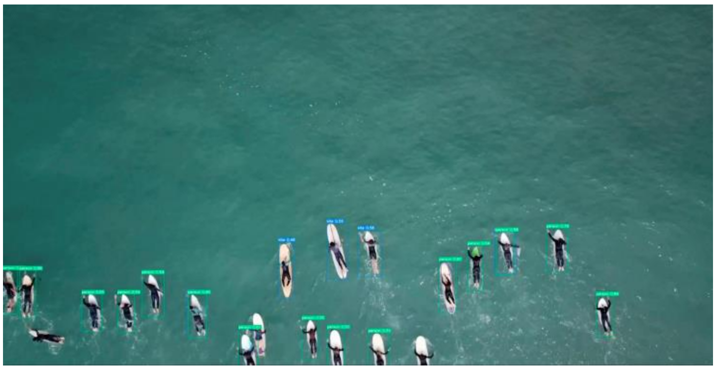

Fig. 1. Human detection by YOLOv7

Figure 2 shows the classification of objects using YOLOv7. As we can see, the YOLOv7 architecture has detected all the people in the frame and highlighted them with a dark blue rectangle with the class "person".

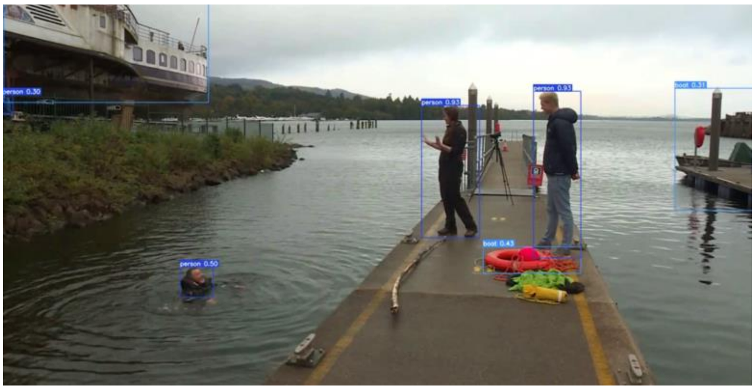

Fig. 2. Object detection by YOLOv7 when human partially submerged on water.

An analysis of "misclassified objects" in the context of people-on-water detection using YOLO architectures revealed that this type of error is an important aspect for optimizing model performance. Misclassified objects can include cases where the model falsely identifies inanimate objects as people or misses people by classifying them as part of the environment. Reducing the number of such errors requires improving training algorithms, increasing the diversity of training data, and optimizing data preprocessing. Figure 3 shows the incorrectly classified "person" object in the frame. Instead of the correct class, YOLO categorized the person as "bird".

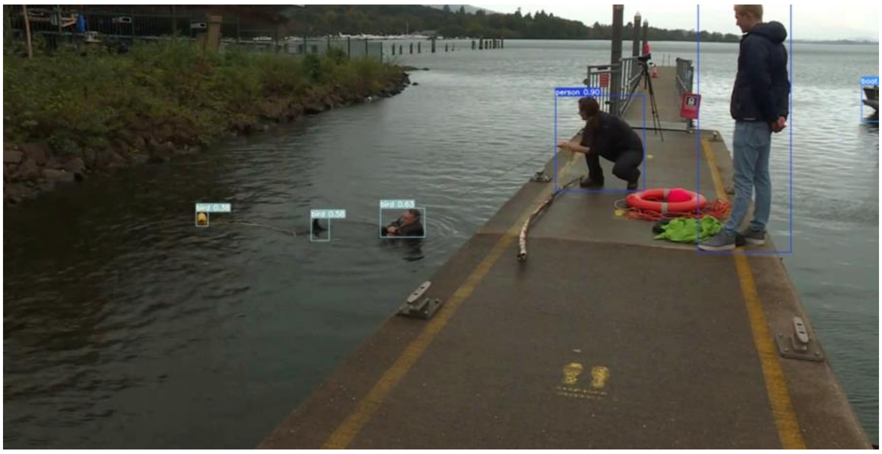

Fig. 3. Misclassified partially submerged human.

In the context of using YOLO for object detection, misclassification can occur due to several factors. Among them is the limited ability of the model to effectively distinguish between objects on complex backgrounds or in conditions of object overlap. Insufficient data diversity can lead to a high specificity of the model, which works well only on data like the training set. Also, the quality of data annotation plays an important role: incorrectly labeled objects in the training set can distort training. In addition, limitations of the YOLO architecture, such as loss of detail due to reduced input image size or suboptimal anchor frame size, can also cause classification errors. To reduce the number of misclassified objects in YOLO architectures, it is recommended to increase the variety and volume of training data, including images from different perspectives and under different lighting conditions. It is important to improve data preprocessing, in particular, the use of augmentation techniques to simulate potential operating conditions. It is also necessary to optimize the architecture, tune hyperparameters, and use techniques to reduce the loss of details on small objects, for example, by introducing more accurate detection mechanisms.

The YOLOv8 architecture proved to be the most effective among all versions of YOLO models in the task of detecting people on water. This version not only showed high accuracy in detecting persons, but was also able to build skeletons on the detected people. This feature greatly expands the analysis capabilities, allowing for further motion analysis and detection of critical states, such as determining whether a person is alive. This approach opens up new horizons for the development of rescue systems and water safety monitoring. Figure 4 shows the result of recognizing a person in the water using YOLOv8. The person was highlighted with a red rectangle and key points were overlaid for further processing and motion analysis. If the detected rectangle with a person moved and changed its coordinates, it assumed that the person was moving and potentially alive.

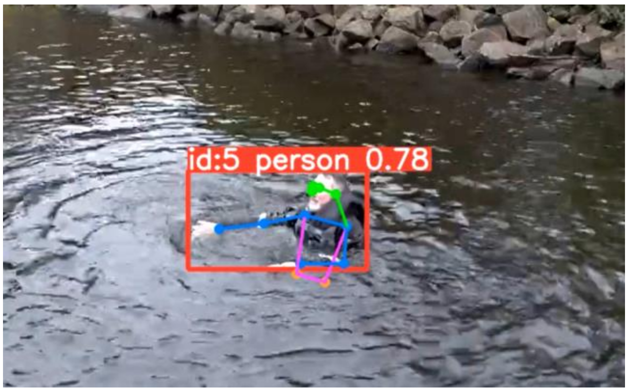

Fig. 4. Object detection when human partially submerged on water.

YOLOv8 excels at detecting multiple people in a video frame, demonstrating the ability to build skeletons quickly and accurately for each detected person. Figure 5 shows the result of capturing several people on the surface of the water with YOLO. After detecting a person in the water, a skeleton was built for each of them to track joint movements.

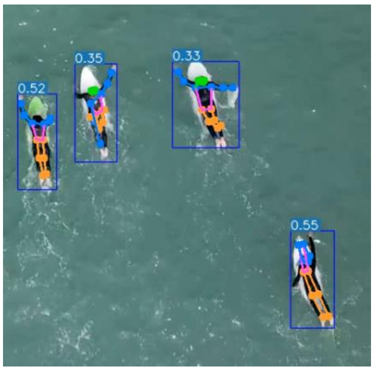

Fig. 5. Multiple human detection by YOLOv8

YOLOv8 is highly efficient in real-time, delivering high frame rates even on devices with limited processing power. This expands its applicability in a wide range of environments and on a variety of devices. A comparative analysis of object detection errors using different versions of the YOLO architecture is performed. This analysis provides a deeper understanding of how each version copes with the challenges of object identification in a variety of environments. By looking at typical errors such as partial detection, misclassification, or complete object misses, we can determine which aspects of the architecture need to be optimized to improve detection accuracy. Table 3 provides a detailed analysis of the errors made by different versions of the YOLO architectures during the human detection experiments. It shows the percentage of cases where objects were partially detected, misclassified, or completely missed by each version of the models from YOLOv3 to YOLOv8.

Table 3. Analysis of detection errors in different versions of YOLO architectures

Architecture	**Partially Detected**	**Misclassified**	**Missed**
YOLOv3	16%	9%	8%
YOLOv4	10%	7%	6%
YOLOv5	11%	8%	6%
YOLOv6	21%	12%	11%
YOLOv7	10%	6%	6%
YOLOv8	8%	4%	4%

It can be concluded that YOLOv8 significantly outperforms the other versions in all respects, indicating its high efficiency and accuracy in detecting people in the water. On the other hand, YOLOv6 showed the highest percentage of errors, which emphasizes the need for further optimization. When the YOLOv8 model fails in challenging scenarios, especially when dealing with low light conditions, occlusion, and different water textures, it can have a significant impact on its performance. Low light conditions pose challenges for any computer vision system due to reduced visibility and contrast between objects and their backgrounds. This can cause YOLOv8 to lose its ability to correctly identify people in the water, as the features the model looks for to classify and localise objects become less distinct. Such conditions can also distort the characteristics of an object, making its silhouette blurry or incomplete. Occlusion presents another serious problem when objects are partially or completely blocked by other objects in the image. This can be particularly common in dynamic water environments where people may be partially obscured by waves or other objects. In such cases, YOLOv8 may misinterpret a person's shape or position, resulting in false detections or misses. Water textures add another layer of complexity, as they can vary significantly from calm water to stormy water with large waves and splashes. Sun glare on the surface of the water or changes in the colour of the water from different angles can create a large number of false positives, where the model may detect objects that are not actually there, or vice versa, fail to detect a person in need of help. The diversity and dynamics of the aquatic environment requires the model to be able to adapt to a wide range of visual conditions, which is a significant challenge.

All of these factors highlight the importance of developing more robust deep learning methods that can better adapt to complex detection environments. Optimising models like YOLOv8 to perform in such conditions is possible by expanding training data, incorporating image enhancement techniques to increase visibility in low light conditions, developing algorithms that better recognise and distinguish objects during occlusion, and using more sophisticated texture analysis techniques to identify people in different water conditions.

9 Discussion

The analysis of the results of the experiment with detecting people on water using different versions of YOLO architectures opens up interesting prospects and challenges for further research. It was found that the improvement of the architecture and training methods from YOLOv3 to YOLOv8 significantly increased the effectiveness of the models in detecting people in the water area. The results of YOLOv8 are especially impressive, which emphasizes the progress in the development of deep neural networks and their application in specific detection tasks. YOLOv8 demonstrated the best results, which indicates significant progress in object detection technology. This version effectively solves the problem of detecting people in the water, demonstrating high accuracy and response. This success can be attributed to improvements in the image processing algorithm, including better recognition of the scene context and optimization to handle a variety of lighting conditions and backgrounds. On the other hand, YOLOv6 showed lower performance, which emphasizes the importance of further optimization and adaptation of models for specific tasks. This opens up a discussion about the need to balance the versatility of models with their specialization for specific use cases. One of the key challenges identified during the experiments is the high computational resource requirements, especially for versions such as YOLOv5. This requires the development of optimization strategies and trade-offs between model performance and the availability of resources for training and implementation. At the same time, the results indicate a significant potential for further improvement of YOLO models through fine-tuning and contextualization. Future research could focus on developing specialized versions of YOLO that are optimized for detecting people in complex environments, such as water areas with a variety of influencing factors. The results of the experiment are important for the development of security and surveillance systems, especially in the context of improving the efficiency of search and rescue operations on the water. Improvements in human detection algorithms can contribute to faster identification of victims and more effective emergency response. In general, the discussion of the results emphasizes the importance of further research in the field of computer vision and the development of intelligent systems capable of adapting to complex and dynamic real-world conditions. The development and optimization of YOLO architectures for specific object detection tasks opens up new opportunities for improving safety and efficiency in various applications.

10 Conclusions

The promising results obtained during the experiments indicate that the YOLOv8 architecture is not only theoretically sound, but also practically applicable. Its high accuracy and low error rate make it an invaluable tool for search and rescue missions, where fast and correct decision-making is of paramount importance. Given its performance, it can make a significant contribution to saving lives in emergencies by effectively locating and identifying people in need of assistance. Thus, the results confirm the effectiveness, reliability, and practicality of the proposed architecture. Due to its high precision, accuracy, recall, and F1-Score, as well as minimal errors, the architecture proves to be better than previous models such as YOLOv3, YOLOv4, YOLOv5, YOLOv6, and YOLOv7. Its practical application in real-world search and rescue missions makes it a significant contribution to the field, opening up prospects for future implementations and developments in emergency response systems. The findings lay a solid foundation for further research and development in this area, paving the way for more advanced and reliable rescue technologies in the future. The modular nature of the technology also means that it can be adapted to different types of disasters and emergencies, from forest fires to earthquakes. In essence, the inherent scalability of this project ensures that as technology evolves, the architecture can be adapted and expanded, making it a long-term solution for disaster response on a global scale.

Acknowledgements

The work was carried out under the auspices of the Software Engineering Department and the research laboratory "Information Technologies in Learning and Computer Vision Systems" at the Kharkiv National University of Radio Electronics. The authors extend their profound appreciation to the scientists from the Technical University of Applied Sciences Wildau, Brandenburg, Germany, and the Volkswagen Foundation for their invaluable support and contributions. Server used for research is med-IoT-KI, TH Wildau, project 85053663 "AI-supported IoT environment for medical examinations" MWFK program to promote the infrastructure for research, development and innovation (InfraFEI) with funds from the European Regional Development Fund (ERDF). This international collaboration has not only paved the way for groundbreaking research but also fostered an enriching academic and investigative exchange among the participating institutions.

References

1. Alwani AA, Chahir Y, Goumidi DE, Molina M, Jouen F (2014) 3D-Posture Recognition Using Joint Angle Representation. In: Laurent A, Strauss O, Bouchon-Meunier B, Yager RR (eds) Information Processing and Management of Uncertainty in Knowledge-Based Systems. Springer International Publishing, Cham, pp 106–115
2. Chen C, Min H, Peng Y, Yang Y, Wang Z (2022) An Intelligent Real-Time Object Detection System on Drones. Applied Sciences 12:10227
3. Gallego A, Pertusa A, Gil P, Fisher RB (2019) Detection of bodies in maritime rescue operations using unmanned aerial vehicles with multispectral cameras. Journal of Field Robotics 36:782–796
4. Maji D, Nagori S, Mathew M, Poddar D (2022) YOLO-Pose: Enhancing YOLO for Multi Person Pose Estimation Using Object Keypoint Similarity Loss.
5. Bilous NV, Ahekian IA, Kaluhin VV (2023) DETERMINATION AND COMPARISON METHODS OF BODY POSITIONS ON STREAM VIDEO. RIC 52
6. Hernández D, Cecilia JM, Cano J-C, Calafate CT (2022) Flood Detection Using Real-Time Image Segmentation from Unmanned Aerial Vehicles on Edge-Computing Platform. Remote Sensing 14:223

7. Li X, Guo Y, Pan W, Liu H, Xu B (2023) Human Pose Estimation Based on Lightweight Multi-Scale Coordinate Attention. Applied Sciences 13:3614
8. Neto JGDS, Teixeira JMXN, Teichrieb V (2020) Analyzing embedded pose estimation solutions for human behaviour understanding. In: Anais Estendidos do Simpósio de Realidade Virtual e Aumentada (SVR Estendido 2020). Sociedade Brasileira de Computação, Brasil, pp 30–34
9. Rakova AO, Bilous NV (2020) REFERENCE POINTS METHOD FOR HUMAN HEAD MOVEMENTS TRACKING. RIC 0:121–128
10. Smith J, Loncomilla P, Ruiz-Del-Solar J (2023) Human Pose Estimation Using Thermal Images. IEEE Access 11:35352–35370
11. M D, S C, C.M. B (2023) Robust human detection system in flood related images with data augmentation. Multimed Tools Appl 82:10661–10679
12. Hramm O, Bilous N, Ahekian I (2019) Configurable Cell Segmentation Solution Using Hough Circles Transform and Watershed Algorithm. In: 2019 IEEE 8th International Conference on Advanced Optoelectronics and Lasers (CAOL). IEEE, Sozopol, Bulgaria, pp 602–605
13. Wu X, Li W, Hong D, Tao R, Du Q (2022) Deep Learning for UAV-based Object Detection and Tracking: A Survey. IEEE Geosci Remote Sens Mag 10:91–124
14. Valarmathi B, Kshitij J, Dimple R, Srinivasa Gupta N, Harold Robinson Y, Arulkumaran G, Mulu T (2023) Human Detection and Action Recognition for Search and Rescue in Disasters Using YOLOv3 Algorithm. Journal of Electrical and Computer Engineering 2023:1–19
15. Tan L, Lv X, Lian X, Wang G (2021) YOLOv4_Drone: UAV image target detection based on an improved YOLOv4 algorithm. Computers & Electrical Engineering 93:107261
16. Dadrass Javan F, Samadzadegan F, Gholamshahi M, Ashatari Mahini F (2022) A Modified YOLOv4 Deep Learning Network for Vision-Based UAV Recognition. Drones 6:160
17. Nguyen H-C, Nguyen T-H, Scherer R, Le V-H (2022) Unified End-to-End YOLOv5-HR-TCM Framework for Automatic 2D/3D Human Pose Estimation for Real-Time Applications. Sensors 22:5419
18. Li C, Li L, Jiang H, et al (2022) YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. https://doi.org/10.48550/ARXIV.2209.02976
19. He C, Saha P (2023) Investigating YOLO Models Towards Outdoor Obstacle Detection For Visually Impaired People. https://doi.org/10.48550/ARXIV.2312.07571
20. Horvat M, Jelečević L, Gledec G (2023) Comparative Analysis of YOLOv5 and YOLOv6 Models Performance for Object Classification on Open Infrastructure: Insights and Recommendations.
21. Yang W, Tang X, Jiang K, Fu Y, Zhang X (2023) An Improved YOLOv5 Algorithm for Vulnerable Road User Detection. Sensors 23:7761
22. Yan J, Zhou Z, Zhou D, Su B, Xuanyuan Z, Tang J, Lai Y, Chen J, Liang W (2022) Underwater object detection algorithm based on attention mechanism and cross-stage partial fast spatial pyramidal pooling. Front Mar Sci 9:1056300
23. Zeng Y, Zhang T, He W, Zhang Z (2023) YOLOv7-UAV: An Unmanned Aerial Vehicle Image Object Detection Algorithm Based on Improved YOLOv7. Electronics 12:3141
24. Chen H, Zhou G, Jiang H (2023) Student Behavior Detection in the Classroom Based on Improved YOLOv8. Sensors 23:8385
25. Wang G, Chen Y, An P, Hong H, Hu J, Huang T (2023) UAV-YOLOv8: A Small-Object-Detection Model Based on Improved YOLOv8 for UAV Aerial Photography Scenarios. Sensors 23:7190
26. Berndt J, Meißner H, Kraft T (2023) ON THE ACCURACY OF YOLOV8-CNN REGARDING DETECTION OF HUMANS IN NADIR AERIAL IMAGES FOR SEARCH

AND RESCUE APPLICATIONS. Int Arch Photogramm Remote Sens Spatial Inf Sci XLVIII-1/W2-2023:139–146

27. Redmon J, Farhadi A (2018) YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767
28. Bochkovskiy A, Wang C-Y, Liao H-YM (2020) YOLOv4: Optimal Speed and Accuracy of Object Detection.
29. Jocher G (2020) Ultralytics YOLOv5. https://doi.org/10.5281/zenodo.3908559
30. Li C, Li L, Geng Y, Jiang H, Cheng M, Zhang B, Ke Z, Xu X, Chu X (2023) YOLOv6 v3.0: A Full-Scale Reloading.
31. Wang C-Y, Bochkovskiy A, Liao H-YM (2022) YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696
32. Jocher G, Chaurasia A, Qiu J (2023) Ultralytics YOLOv8.
33. Lin T-Y, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, Perona P, Ramanan D, Zitnick CL, Dollár P (2014) Microsoft COCO: Common Objects in Context. https://doi.org/10.48550/ARXIV.1405.0312
34. Sambolek S, Ivasic-Kos M (2021) SEARCH AND RESCUE IMAGE DATASET FOR PERSON DETECTION - SARD. https://doi.org/10.21227/AHXM-K331
35. Varga LA, Kiefer B, Messmer M, Zell A (2021) SeaDronesSee: A Maritime Benchmark for Detecting Humans in Open Water. https://doi.org/10.48550/ARXIV.2105.01922