COMPUTER SCIENCES AND INFORMATION TECHNOLOGIES

UDC 004.032.26(045)
DOI:10.18372/1990-5548.79.18429

[1]**V. M. Sineglazov,**
[2]**M. V. Savchenko**

A COMPREHENSIVE FRAMEWORK FOR UNDERWATER OBJECT DETECTION BASED ON IMPROVED YOLOv8

[1]Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine
[2]Educational and scientific Institute for Applied System Analysis, National Technical University of Ukraine "Ihor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine
E-mails: [1]svm@nau.edu.ua ORCID 0000-0002-3297-9060, [2]mykhailo_savchenko@outlook.com

Abstract—*Underwater object detection poses unique challenges due to issues such as poor visibility, small densely packed objects, and target occlusion. In this paper, we propose a comprehensive framework for underwater object detection based on improved YOLOv8, addressing these challenges and achieving superior performance. Our framework integrates several key enhancements including Contrast Limited Adaptive Histogram Equalization for image preprocessing, a lightweight GhostNetV2 backbone, Coordinate Attention mechanism, and Deformable ConvNets v4 for improved feature representation. Through experimentation on the UTDAC2020 dataset, our model achieves 82.35% precision, 80.98 % recall, and 86.21 % mean average precision at IoU = 0.5. Notably, our framework outperforms the YOLOv8s model by a significant margin, while also being 15.1% smaller in terms of computational complexity. These results underscore the efficiency of our proposed framework for underwater object detection tasks, demonstrating its potential for real-world applications in underwater environments.*

Index Terms—Underwater object detection; classification problem; YOLO; hybrid neural networks; deep learning.

I. INTRODUCTION

The utilization of automated intelligent underwater vehicles in ocean exploration has proven to be instrumental in various domains such as marine life exploration, fisheries, ecological monitoring, and military applications. The integration of deep learning, computer vision, and object detection algorithms has marked a significant leap forward in advancing the capabilities of these underwater vehicles. However, the realm of underwater image detection confronts a myriad of unique challenges. Issues such as poor image quality, the presence of small and densely-packed targets that are challenging to discern, a scarcity of high-quality training datasets, and the constrained computational power of underwater vehicle hardware collectively impede the seamless deployment of efficient detection algorithms. This underscores the need for innovative approaches specifically tailored to enhance the precision and efficiency of underwater object detection. Given the critical role that automated underwater vehicles play in various applications ranging from marine biology research to underwater infrastructure inspection, overcoming these challenges is imperative for unlocking their full potential and facilitating groundbreaking discoveries in the depths of our oceans. As such, this article delves into the current landscape of underwater object detection methodologies, highlighting both advancements and existing limitations, while also proposing a new deployment-efficient framework for this task (Fig. 1).

Fig. 1. Sample from UTDAC2020 dataset demonstrating unique challenges in underwater object detection tasks

II. RELATED WORK

In the realm of object detection, while algorithms have demonstrated proficiency on generic datasets, their application to underwater scenes poses distinct challenges. These challenges stem from the inherent complexities of underwater environments, including poor image quality, color distortion, light interference, and the prevalence of small, densely-packed targets. Consequently, the task of UOD necessitates a nuanced approach, typically bifurcated into image pre-processing and object detection subtasks.

A. Image preprocessing

Underwater visibility challenges stem from water molecules and suspended particles affecting light distortion and color absorption. Recent advancements include leveraging absorption differences in color wavelengths for transmission plot estimation, employing graph tangent theory to fortify underwater neural networks [1], and addressing factors like absorption, scattering, and color distortion in underwater imagery [2]. Methods for underwater image dehazing and color restoration have been proposed [3], along with correction techniques based on polarization imaging models [4]. Algorithms such as dark channel prior [5], Retinex-model-based decomposition [6], and color correction within HSV and Retinex models [7] have also been applied to solve the aforementioned challenges. The twin adversarial contrastive learning algorithm has been developed for underwater image preprocessing [8].

A comprehensive framework for underwater image enhancement covered in [9] combines techniques like image fusion, edge sharpening, and contrast enhancement. Utilizing algorithms like MSRCR, CLAHE, and homomorphic filtering within different color spaces improves color saturation and contrast, resulting in significant quality enhancement. Augmentation methods enhance domain generalization in underwater object detection [10], while candidate frame fusion algorithms refine underwater target detection [11]. Transfer learning techniques have led to exceptional object identification results in low-quality underwater videos in [12], achieving an average classification accuracy of 99.68% for 23 fish species.

B. Object detection and classification

In the realm of deep-learning-based object detection models, two main methodologies have emerged: anchor-based and anchor-free algorithms. Anchor-based approaches like Faster R-CNN [13], SSD [14], and RetinaNet [15] rely on predefined anchor boxes for object localization. In contrast, anchor-free algorithms such as YOLOX [16] and FCOS [17] calculate only the center point and position coordinates of bounding boxes, simplifying detection. Recent advancements, including attention mechanisms, have further improved object localization and classification accuracy.

Effective object detection in underwater environments requires advanced techniques due to challenges posed by small and dense targets. Deep Convolutional Neural Networks (CNNs), particularly the YOLO series [18], excel in such tasks. Researchers have applied YOLO-based architectures for various underwater applications, showcasing their versatility. For instance, a real-time YOLO-based CNN achieved a 93 % fish detection accuracy [19]. YOLOv2 and YOLOv3 were utilized for marine-animal detection [20], and a lightweight underwater object detection framework based on YOLOv4 was introduced [21]. Other researchers have proposed novel approaches like TC-YOLO [22], mDFLAM [23], SA-FPN [24]. These algorithms address challenges such as small target detection, noisy samples, and mutual occlusion.

Despite advancements, current methodologies have drawbacks. Many focus on isolated aspects of the detection pipeline, neglecting comprehensive approaches that synergize image preprocessing and detection algorithms. Moreover, algorithms often overlook practical constraints imposed by hardware limitations of underwater vehicles, hindering scalability and applicability. There's a need for holistic, hardware-aware approaches to underwater object detection.

III. PROBLEM STATEMENT

Generally, loss function and weight update procedure for object detection and classification tasks can be defined as:

$$L = \lambda_{\text{coord}} \cdot L_{\text{coord}} + \lambda_{\text{conf}} \cdot L_{\text{conf}} + \lambda_{\text{class}} \cdot L_{\text{class}},$$

where $L_{\text{coord}}, L_{\text{conf}}, L_{\text{class}}$ are the localization, confidence and classification losses in that order, $\lambda_{\text{coord}}, \lambda_{\text{conf}}, \lambda_{\text{class}}$ are the coefficients to balance the influence of each component in general loss function. The velocity term is defined as:

$$V^t = \beta V^{t-1} + \nabla_\theta L\left(\theta^{t-1}\right),$$

where β is the momentum, and the weight update rule takes the following form:

$$\theta^t = \theta^{t-1} - \eta V^t,$$

where η is the learning rate.

Specifically, for YOLOv8, the loss function can be defined as follows:

$$L = \frac{\lambda_{\text{box}}}{N_{\text{pos}}} \sum_{x,y} \mathbf{1}_{c^*_{x,y}} \left[1 - q_{x,y} + \frac{\left\| b_{x,y} - \hat{b}_{x,y} \right\|_2^2}{\rho^2} + \alpha_{x,y} v_{x,y} \right]$$
$$+ \frac{\lambda_{\text{clf}}}{N_{\text{pos}}} \sum_{x,y} \sum_{c \in \text{classes}} y_c \log(\hat{y}_c) + (1 - y_c) \log(1 - \hat{y}_c)$$
$$+ \frac{\lambda_{\text{dfl}}}{N_{\text{pos}}} \sum_{x,y} \mathbf{1}_{c^*_{x,y}} \left[-\left(q_{(x,y)+1} - q_{x,y}\right) \log\left(\hat{q}_{x,y}\right) \right.$$
$$\left. + \left(q_{x,y} - q_{(x,y)-1}\right) \log\left(\hat{q}_{(x,y)+1}\right) \right],$$

where

$$q_{x,y} = \text{IoU}_{x,y} = \frac{\hat{\beta}_{x,y} \cap \beta_{x,y}}{\hat{\beta}_{x,y} \cup \beta_{x,y}},$$

$$v_{x,y} = \frac{4}{\pi^2} \left(\arctan\left(\frac{w_{x,y}}{h_{x,y}} \right) - \arctan\left(\frac{\hat{w}_{x,y}}{\hat{h}_{x,y}} \right) \right)^2,$$

$$\alpha_{x,y} = \frac{v}{1 - q_{x,y}},$$

$$\hat{y}_c = \sigma(\cdot),$$

$$\hat{q}_{x,y} = \text{softmax}(\cdot).$$

Here, N_{pos} represents the number of cells featuring an object, $\mathbf{1}_{c^*_{x,y}}$ is an indicator function for the cells featuring an object, $\beta_{x,y}$ is the ground truth bounding box position, $b_{x,y}$ is the predicted box of the respective cell, $\hat{\beta}_{x,y}$ are the coordinates of the center point of the ground truth bounding box, y_c represents the ground truth label for class c for each individual grid cell (x, y) in the input, $q_{(x,y)+1}$ are the nearest left and right predicted boxes IoU which belong to $c^*_{x,y}$, $w_{x,y}$ and $h_{x,y}$ are width and height of the box, ρ is the diagonal length of the smallest enclosing box covering the predicted and ground truth boxes.

The best candidate is then determined by each cell for predicting the bounding box of the object. In YOLOv8, CIoU [25] is used as the box loss, binary cross entropy is used for multi-label classification as the classification loss and distribution focal loss [26] is used as the third term in general loss function.

YOLOv8, considered a state-of-the-art model for object detection, offers significant improvements over earlier versions, featuring a lighter backbone structure and a decoupled head design. It introduces anchor-free design and incorporates distribution focal loss and task-aligned label matching in its loss function. Despite offering slimmed-down versions like YOLOv8s, YOLOv8 still faces challenges such as high computational complexity and network transmission volume, hampering its speed and hardware requirements. A more hardware-aware approach is needed to address these issues and enable efficient operation on edge devices.

IV. Problem Solution

To solve the unique challenges, present in UOD task, such as visibility issues, the presence of small densely packed objects and target occlusion, we present a new method consisting of three main advancements: image preprocessing module using Contrast Limited Adaptive Histogram Equalization (CLAHE), replacing YOLOv8 pre-packaged Darknet-53 backbone with lightweight GhostNetV2 backbone, incorporating Coordinate Attention (CA) to highlight regions of interest (RoI) within the image, and replacing in-built convolutional layers in YOLOv8 neck and C2f block bottleneck with Deformable ConvNets v4 (DCNv4).

A. Preprocessing module

As confirmed in our ablation experiments, CLAHE [27] introduces a boost in recall by improving the details and restoring original colors of the underwater image, which results in a decreased number of missed detections when it comes to smaller targets, CLAHE algorithm consists of four steps:

1) Divide the image into non-overlapping tiles of a specified size.
2) Compute the histogram $H_i(k)$ for each tile i.
3) Perform histogram equalization independently for each tile:

$$I_{eq}(x,y) = L\left(I(x,y), H_i\right),$$

where L is the function that maps the intensity values of $I(x,y)$ to the corresponding equalized values using the histogram H_i.

4) Apply contrast limiting:

$$I_{\text{clahe}}(x,y) = \begin{cases} I_{\text{eq}}(x,y), \\ I_{\text{eq}}(x,y) + \dfrac{I_{\text{eq}}(x,y) - \text{clipLimit}}{N_i}. \end{cases}$$

The images from training and testing set were resized to 640x640 and then processed by CLAHE algorithm using OpenCV library with ‘clip limit’ value manually lowered to 2 to avoid unnecessary changes in color. Average processing time for each image sample is 2.1 ms, which we consider an acceptable trade-off between time and performance (Fig. 2).

Fig. 2. Image taken from UTDAC2020 dataset before and after preprocessing

B. Lightweight network backbone

To achieve higher inference speed on edge devices, we replaced the YOLOv8 DarkNet-53 backbone with the lightweight GhostNetV2 backbone [28]. The GhostNetV2 architecture reduces network parameter count and computational cost by utilizing 1x1 convolutions to aggregate local and long-range information with decoupled fully connected (DFC) attention. The GhostNet building block comprises Ghost modules, replacing standard convolutions, where intrinsic features are generated using $Y' = X * F_{1\times1}$ with $*$ being convolution operation, $F_{1\times1}$ being point-wise convolution and Y' being intrinsic features with the size lower than the original output feature size. Then depth-wise convolutions are used to generate more features based on obtained intrinsic features, and the output feature Y by concatenation along the channel dimension as $Y = Concat\left(\left[Y', Y' * F_{dp}\right]\right)$, where F_{dp} is 1x1 convolutional filter and Y is the output feature.

To compensate for lowered representation ability of the module, GhostNetV2 uses DFC attention module. A straightforward usage of fully-connected layers to generate attention map is considered deployment inefficient, and thus replaced by the operation of decomposing it into two fully-connected layers and aggregating features along horizontal and vertical dimensions, effectively reducing the computation costs. Then, the input feature is sent to two branches of the network, which are Ghost and DFC modules, and the final output is calculated as element-wise product of normalized attention map and the feature. Notably, to further reduce the computational cost of DFC operation, the feature is down-sampled by both dimensions. Overall, the usage of GhostNetV2 allows for the reduction of the parameter count of the backbone, making it more deployment-efficient while maintaining accuracy.

C. Coordinate attention

Coordinate Attention (CA) [36], enhances spatial information utilization within feature maps by learning attention weights for each spatial position. Unlike traditional channel-wise attention mechanisms, CA focuses on relationships between spatial positions, aiming to capture long-range dependencies and contextual information. Firstly, direction-aware feature maps are generated for each spatial dimension, concatenated, and passed through a 1x1 convolutional function F_1. Next, 1x1 convolutional transformations F_h and F_w are applied, with their outputs serving as attention weights for spatial dimensions. The final block output is computed as $y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j)$. Placing the CA block in object detection algorithms enhances performance by enabling the model to focus on relevant spatial positions, improving the detection of smaller targets and reducing false negatives.

D. Deformable Convolutions

Deformable convolutions are adaptive convolutional operations that adjust the receptive field of each kernel based on input data, enhancing the network's ability to capture spatial structures and patterns, especially for objects at different scales or positions. Unlike traditional convolutions with fixed receptive fields, deformable convolutions introduce learnable offsets, enabling kernels to sample from arbitrary locations in the input feature map [29].

In deformable convolutions, the regular grid $R = (-1,-1),(-1,0),...,(0,1),(1,1)$ is augmented with offsets Δp_n, yielding a feature map Y at position p_0 as:

$$Y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n + \Delta p_n),$$

where x represents the input feature map, w are the weights, p_n are the coordinates in the grid R, and Δp_n are the offsets.

Another extension, Deformable RoI Pooling, an extension of traditional RoI pooling used in object detection, extracts fixed-size feature maps from varying-sized feature maps generated by a CNN.

The latest version of Deformable Convolution operator [30] called DCNv4 features further speed-up and optimizations of memory access by minimizing the count of redundant operations. In our work, DCNv4 is used in both YOLOv8 neck part and C2f bottleneck, demonstrating significant boost in small target detection capabilities of the model.

V. EXPERIMENTS

A. Dataset overview

A challenging underwater detection dataset UTDAC2020, which is short for Underwater Target

Detection Algorithm Competition 2020, has been selected to test the performance of the proposed algorithm. The dataset features 5168 training and 1293 validation images in various resolutions (3840x2160, 1920x1080, 720x405 and 586x480), four classes are represented (echinus, holothurian, scallop and starfish).

B. Implementation details

The experimental setup consisted of Intel Xeon E5 CPU (2.00 GHz), two NVIDIA Tesla T4 GPU with 16 GB VRAM each with Ubuntu 20.04.6 LTS, Python 3.10.13, CUDA 12.1, PyTorch 2.2.1 installed. The training process was limited to 200 epochs with early stopping, batch size was fixed at 32, stochastic gradient descent has been used as an optimization algorithm with momentum 0.95 and weight decay coefficient 0.005, initial learning rate set to 0.01. Default augmentation strategies from YOLOv8 have been applied, and no other augmentations have been used.

C. Experiment results

The following metrics have been used to assess the performance of the algorithm:

- precision, defined as true positives count, divided by the sum of true positives and false positives, indicating false-detection rate of the algorithm;
- recall, defined as true positives count, divided by the sum of true positives and false negatives, reflecting the missed-detection rate of the algorithm;
- $mAp^{IoU=0.5}$, defined as the mean average precision (mAp) for all target classes across entire dataset with IoU = 0.5 set as an evaluation threshold;
- floating point operations count, measured in GFLOPs, reflecting the computational complexity of the network.

The results of the experiments are as follows (Table I).

TABLE I THE RESULTS OF THE EXPERIMENTS

Model	**Precision**	**Recall**	**$mAp^{IoU=0.5}$**	**GFLOPs**
YOLOv8n	71.02%	66.92%	82.65%	**8.1**
YOLOv8s	75.02%	69.78%	84.71%	28.4
YOLOv8m	76.72%	70.53%	84.92%	78.7
YOLOv8l	79.24%	73.12%	85.09%	164.8
Ours	**82.35%**	**80.98%**	**86.21%**	24.1

The proposed models surpassed even larger YOLOv8l model in precision, recall and $mAp^{IoU = 0.5}$ metrics, while maintaining smaller size, acceptable to be deployed on devices with limited processing power.

VI. CONCLUSIONS

This paper proposes a comprehensive framework tailored for Underwater Object Detection, enhancing YOLOv8 architecture to improve detection performance in underwater environments. Key enhancements include a novel image preprocessing module using Contrast Limited Adaptive Histogram Equalization to address visibility issues and enhance object detection accuracy. The substitution of the Darknet-53 backbone with the lightweight GhostNetV2 backbone reduces computational overhead while maintaining or improving accuracy. Incorporating a Coordinate Attention mechanism highlights regions of interest, and replacing convolutional layers with Deformable ConvNets v4 enhances adaptability to non-rigid underwater environments. The framework achieves superior computational efficiency compared to pre-made slimmed versions of YOLO, demonstrating an optimal balance for real-world underwater object detection applications.

REFERENCES

[1] F. Alenezi, A. Armghan, and K. C. Santosh, “Underwater image dehazing using global color features,” *Engineering Applications of Artificial Intelligence*, vol. 116, p. 105489, Nov. 2022, https://doi.org/10.1016/j.engappai.2022.105489.

[2] K. Hu, C. Weng, Y. Zhang, J. Jin, and Q. Xia, “An Overview of Underwater Vision Enhancement: From Traditional Methods to Recent Deep Learning,” *Journal of Marine Science and Engineering*, vol. 10, no. 2, p. 241, Feb. 2022, https://doi.org/10.3390/jmse10020241.

[3] M. Han, Z. Lyu, T. Qiu, and M. Xu, “A Review on Intelligence Dehazing and Color Restoration for

Underwater Images," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 5, pp. 1820–1832, May 2020, https://doi.org/10.1109/tsmc.2017.2788902.

[4] H. Hu, L. Zhao, B. Huang, X. Li, H. Wang, and T. Liu, "Enhancing Visibility of Polarimetric Underwater Image by Transmittance Correction," *IEEE Photonics Journal*, vol. 9, no. 3, pp. 1–10, Jun. 2017, https://doi.org/10.1109/jphot.2017.2698000.

[5] Kaiming He, Jian Sun, and Xiaoou Tang, "Single Image Haze Removal Using Dark Channel Prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011, https://doi.org/10.1109/tpami.2010.168.

[6] X. Fu, P. Zhuang, Y. Huang, Y. Liao, X.-P. Zhang, and X. Ding, "A retinex-based enhancing approach for single underwater image," *International Conference on Image Processing*, Oct. 2014, https://doi.org/10.1109/icip.2014.7025927.

[7] W.-H. Zhang, G. Li, and Z. Ying, "A new underwater image enhancing method via color correction and illumination adjustment," *Visual Communications and Image Processing*, Dec. 2017, https://doi.org/10.1109/vcip.2017.8305027.

[8] R. Liu, Z. Jiang, S. Yang, and X. Fan, "Twin Adversarial Contrastive Learning for Underwater Image Enhancement and Beyond," *IEEE transactions on image processing*, vol. 31, pp. 4922–4936, Jan. 2022, https://doi.org/10.1109/tip.2022.3190209.

[9] M. Zhang, S. Xu, W. Song, Q. He, and Q. Wei, "Lightweight Underwater Object Detection Based on YOLO v4 and Multi-Scale Attentional Feature Fusion," *Remote Sensing*, vol. 13, no. 22, p. 4706, Nov. 2021, https://doi.org/10.3390/rs13224706.

[10] H. Liu, P. Song, and R. Ding, "WQT and DG-YOLO: towards domain generalization in underwater object detection," *arXiv (Cornell University)*, Apr. 2020, https://doi.org/10.48550/arxiv.2004.06333.

[11] W. Lin, J.-X. Zhong, S. Liu, T. Li, and G. Li, "ROIMIX: Proposal-Fusion Among Multiple Images for Underwater Object Detection," *arXiv (Cornell University)*, May 2020, https://doi.org/10.1109/icassp40776.2020.9053829.

[12] X. Sun et al., "Transferring deep knowledge for object recognition in Low-quality underwater videos," *Neurocomputing*, vol. 275, pp. 897–908, Jan. 2018, https://doi.org/10.1016/j.neucom.2017.09.044.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, https://doi.org/10.1109/tpami.2016.2577031.

[14] W. Liu et al., "SSD: Single Shot MultiBox Detector," *Computer Vision – ECCV 2016*, vol. 9905, pp. 21–37, 2016, https://doi.org/10.1007/978-3-319-46448-0_2.

[15] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018, https://doi.org/10.1109/tpami.2018.2858826.

[16] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO Series in 2021," *Computer Vision and Pattern Recognition (cs.CV)*, Jul. 2018, https://doi.org/10.48550/arXiv.2107.08430.

[17] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-stage Object Detection," *Computer Vision and Pattern Recognition (cs.CV)*, Sep. 2019, https://doi.org/10.48550/arXiv.1904.01355.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," arXiv.org, Jun. 2015, https://doi.org/10.48550/arXiv.1506.02640.

[19] M. Sung, S.-C. Yu, and Y. Girdhar, "Vision based real-time fish detection using convolutional neural network," *OCEANS 2017 – Aberdeen*, Jun. 2017, https://doi.org/10.1109/oceanse.2017.8084889.

[20] M. Pedersen, Joakim Bruslund Haurum, R. Gade, and T. B. Moeslund, "Detection of Marine Animals in a New Underwater Dataset with Varying Visibility," *Computer Vision and Pattern Recognition*, pp. 18–26, Jun. 2019.

[21] M. Zhang, S. Xu, W. Song, Q. He, and Q. Wei, "Lightweight Underwater Object Detection Based on YOLO v4 and Multi-Scale Attentional Feature Fusion," *Remote Sensing*, vol. 13, no. 22, p. 4706, Nov. 2021, https://doi.org/10.3390/rs13224706.

[22] K. Liu, L. Peng, and S. Tang, "Underwater Object Detection Using TC-YOLO with Attention Mechanisms," *Sensors*, vol. 23, no. 5, p. 2567, Jan. 2023, https://doi.org/10.3390/s23052567.

[23] X. Shen, X. Sun, H. Wang, and X. Fu, "Multi-dimensional, multi-functional and multi-level attention in YOLO for underwater object detection," *Neural Computing and Applications*, vol. 35, no. 27, pp. 19935–19960, Jul. 2023, https://doi.org/10.1007/s00521-023-08781-w.

[24] F. Xu, H. Wang, J. Peng, and X. Fu, "Scale-aware feature pyramid architecture for marine object detection," *Neural Computing and Applications*, vol. 33, no. 8, pp. 3637–3653, Jul. 2020, https://doi.org/10.1007/s00521-020-05217-7.

[25] X. Li et al., "Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense

Object Detection," Jun. 2020, https://doi.org/10.48550/arXiv.2006.04388.

[26] S. M. Pizer, R. E. Johnston, J. P. Ericksen, B. C. Yankaskas, and K. E. Muller, "Contrast-limited adaptive histogram equalization: speed and effectiveness," in IEEE Xplore, May 1990, pp. 337–345. https://doi.org/10.1109/VBC.1990.109340.

[27] Y. Tang, K. Han, J. Guo, C. Xu, C. Xu, and Y. Wang, "GhostNetV2: Enhance Cheap Operation with Long-Range Attention," Nov. 2022, https://doi.org/10.48550/arxiv.2211.12905.

[28] Q. Hou, D. Zhou, and J. Feng, "Coordinate Attention for Efficient Mobile Network Design," Mar. 2021, https://doi.org/10.48550/arXiv.2103.02907.

[29] J. Dai et al., "Deformable Convolutional Networks," Jun. 2017, https://doi.org/10.48550/arXiv.1703.06211.

[30] Y. Xiong et al., "Efficient Deformable ConvNets: Rethinking Dynamic and Sparse Operator for Vision Applications," arXiv (Cornell University), Jan. 2024, https://doi.org/10.48550/arxiv.2401.06197.

Received January 16, 2024

Sineglazov Victor. ORCID 0000-0002-3297-9060. Doctor of Engineering Science. Professor. Head of the Department of Aviation Computer-Integrated Complexes.
Faculty of Air Navigation Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine.
Education: Kyiv Polytechnic Institute, Kyiv, Ukraine, (1973).
Research area: Air Navigation, Air Traffic Control, Identification of Complex Systems, Wind / Solar power plant, artificial intelligence.
Publications: more than 700 papers.
E-mail: svm@nau.edu.ua

Savchenko Mykhailo. Master's degree student.
Artificial Intelligence Department, Educational and scientific Institute Applied System Analysis, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute," Kyiv, Ukraine
Education: Kyiv National Linguistic University, Kyiv, Ukraine, (2023)
Research interests: computer vision, natural language processing, artificial intelligence.
E-mail: mykhailo_savchenko@outlook.com

В. М. Синєглазов, М. В. Савченко. Мережа для виявлення підводних об'єктів з використанням модифікованої архітектури YOLOv8

В даній роботі розроблено нейронну мережу для виявлення підводних об'єктів на основі модифікованої архітектури YOLOv8. Розглянуто використання модуля попередньої обробки зображень на основі контрастно-обмеженого адаптивного вирівнювання гістограми, архітектури GhostNetV2 для ефективного вилучення ознак і зменшення загальної кількості параметрів, механізму уваги Coordinate Attention та оператора Deformable ConvNets v4 для покращеної репрезентації ознак. Модель перевірено на вибірці UTDAC2020 (результати – precision 82.35%, recall 80.98%, mAp 86.21% при значенні IoU = 0.5), що випереджає результати YOLOv8s на даній вибірці при зменшенні обчислювальної складності на 15.1%. Результат даної роботи можна застосувати для розробки програмного забезпечення для безпілотних підводних апаратів.

Ключові слова: виявлення підводних об'єктів; класифікація; YOLO; гібридні нейронні мережі; глибоке навчання.

Синєглазов Віктор Михайлович. ORCID 0000-0002-3297-9060. Доктор технічних наук. Професор. Завідувач кафедри авіаційних комп'ютерно-інтегрованих комплексів.
Факультет аеронавігації, електроніки і телекомунікацій, Національний авіаційний університет, Київ, Україна.
Освіта: Київський політехнічний інститут, Київ, Україна, (1973).
Напрям наукової діяльності: аеронавігація, управління повітряним рухом, ідентифікація складних систем, вітроенергетичні установки, штучний інтелект.
Кількість публікацій: більше 700 наукових робіт.
E-mail: svm@nau.edu.ua

Савченко Михайло Володимирович. Студент магістр.
Кафедра штучного інтелекту, Навчально-науковий Інститут прикладного системного аналізу, Національний технічний університет України «Київський політехнічний імені Ігоря Сікорського», Київ, Україна.
Освіта: Київський національний лінгвістичний університет, Київ, Україна, (2023).
Напрям наукової діяльності: комп'ютерний зір, обробка природної мови, штучний інтелект.
E-mail: mykhailo_savchenko@outlook.com