DOI: 10.53469/jtpes.2024.04(06).01

Advancing Automated Surveillance: Real-Time Detection of Crown-of-Thorns Starfish via YOLOv5 Deep Learning

Guokun Xu[1], Ying Xie[2], Yang Luo[3], Yibo Yin[4], Zhengning Li[5], Zibu Wei[6]

[1]Computer Science, Beijing Foreign Studies University, Beijing, China
[2]Computer Science, San Francisco Bay University, Fremont, USA
[3]Computer Science, China CITIC Bank Software Development Center, Beijing, China
[4]Computer Science, Contemporary Amperex Technology USA Inc, Auburn Hills, USA
[5]Computer Science, Georgetown University, Washington, D.C. USA
[6]Computer Science, University of California, Los Angeles, Los Angeles, USA
[1]nicolashsu@hotmail.com, [2]floraxlr999@gmail.com, [3]luoyangdxx@163.com, [4]epark00007@gmail.com, [5]zl132@georgetown.edu, [6]zibuwei@ucla.edu

Abstract: ***The Great Barrier Reef faces significant threats from crown-of-thorns starfish (COTS), which consume coral polyps and contribute to reef degradation. Traditional methods for detecting these starfish are manual and labor-intensive, limiting their scalability and efficiency. This study proposes a real-time detection system using deep learning and computer vision to identify COTS in underwater video frames. We utilize the YOLOv5 model, known for its speed and accuracy in object detection tasks. Extensive data augmentation techniques are employed to handle the challenges of the underwater environment, such as varying lighting conditions and water turbidity. Additionally, we modify the YOLOv5 architecture to improve the detection of small objects like COTS, which often blend into the reef. To enhance detection consistency, we integrate a video object tracking system that maintains object continuity across frames, reducing false positives. Our approach demonstrated significant improvements in detection accuracy, achieving a Public Leaderboard score of 0.715, which places us in the top 2\% of submissions. This highlights the potential of our method for scalable and effective monitoring of the Great Barrier Reef, contributing to conservation efforts by providing a tool for continuous and automated detection of harmful species like COTS.***

Keywords: Great Barrier Reef, Crown-of-Thorns Starfish, YOLOv5, Object Detection, Deep Learning, Marine Conservation.

1. INTRODUCTION

The Great Barrier Reef, the world's largest coral reef system, is a vital marine habitat that supports a diverse array of marine life and contributes significantly to global biodiversity. However, this iconic ecosystem is facing unprecedented threats from climate change, pollution, and particularly, the predatory crown-of-thorns starfish (COTS). COTS are notorious for their voracious appetite for coral polyps, which can lead to large-scale coral loss and degradation of the reef structure. The proliferation of COTS poses a severe risk to the ecological balance and health of the reef, necessitating urgent and effective management interventions.

Traditionally, the detection and monitoring of COTS have relied on manual surveys conducted by divers. These methods are time-consuming, labor-intensive, and limited in scope, making it challenging to cover the vast expanse of the reef effectively. Moreover, manual inspections are prone to human error and subjectivity, leading to inconsistent and incomplete data collection. Given the scale of the threat posed by COTS, there is a critical need for automated and scalable solutions that can provide accurate and real-time monitoring of these starfish across large areas of the reef.

In recent years, advancements in computer vision and deep learning have opened new possibilities for automating the detection of marine species. Object detection models, particularly those based on convolutional neural networks (CNNs), have demonstrated remarkable success in various applications, from autonomous driving to medical imaging. Among these models, the YOLO (You Only Look Once) family has emerged as a leading approach due to its ability to perform real-time object detection with high accuracy.

This study focuses on leveraging the YOLOv5 model, a state-of-the-art object detection framework, to detect

COTS in underwater video frames. YOLOv5 offers several advantages, including fast inference speed, high detection accuracy, and the ability to detect objects of various sizes. However, applying YOLOv5 to underwater imagery presents unique challenges, such as varying lighting conditions, water turbidity, and the small size of COTS relative to the surrounding coral.

In the following sections, we will review related work in the field, analyze the data used for model training, describe the methodology in detail, present experimental results, and discuss the implications of our findings for marine conservation and future research.

2. RELATED WORK

The detection and monitoring of marine life using computer vision and deep learning techniques have garnered significant research interest in recent years. This section provides an overview of related work in the field, focusing on underwater object detection, the application of YOLO models, and the specific challenges associated with detecting small marine organisms like crown-of-thorns starfish (COTS).

S Fayaz et al [1] study proposes a multi-scale convolutional neural network (CNN) for detecting objects in underwater environments. The approach uses feature maps of different scales to effectively detect and identify target objects under complex underwater conditions.J Redmon et al.[2]introduces the YOLO model, which achieves real-time object detection by processing the image in a single network pass, providing a significant speed advantage over traditional methods.

A Bochkovskiy et al.[3]presents YOLOv4, which improves object detection performance with features such as mish activation and mosaic data augmentation, making it suitable for various tasks including complex underwater scenes.Y Lu and G Lu[4] proposes a robust method for feature detection and matching in thermal images, outperforming existing techniques.

AM Rekavandi et al.[5]focuses on enhancing small object detection in underwater videos using a multi-scale approach to capture fine details, critical for identifying small marine organisms like COTS.X Liang et al.[6]propose a hybrid model combining YOLO with attention mechanisms to improve the detection of small objects in cluttered underwater environments, addressing challenges such as occlusion and varying light conditions.Z Lin et al.[7] develops an AI-based model for enhancing robotic control and navigation.wang et al.[8]uses YOLOv3 for detecting fish in aquaculture environments, showcasing the model's utility in monitoring fish populations and health, and highlighting its potential for broader applications in marine life monitoring.

Z Chen et al.[9] develops a real-time underwater monitoring system combining YOLO with video tracking to detect and track marine species, demonstrating the system's effectiveness in continuous marine monitoring.H Chen et al.[10]uses multi-task learning to monitor tool surface changes in ultrasonic welding for quality control.B Alawode et al.[11]addresses underwater image enhancement using deep learning techniques to improve object visibility, which is crucial for effective detection and monitoring in underwater environments.Y Lu[12] proposes unsupervised depth estimation from a single thermal image, achieving accurate results in low-light conditions.
L Xie et al.[13] explores deep learning-based dehazing techniques to reduce the impact of water turbidity on underwater object detection accuracy, enhancing the clarity of underwater images for better detection.

K Liu et al.[14] applies various data augmentation techniques, such as rotation, flipping, and color jittering, to improve the performance of YOLO-based underwater object detection models, enhancing their robustness to diverse underwater conditions.RAIN enhances black-box domain adaptation by preventing overfitting through improved input and network regularization[15].Guyuan Tian and Yuanyuan Xu's[16] study explores designing Chinese characters in Tangut style, blending aesthetics and structure S Song et al.[17] presents a framework for generating synthetic data to augment limited underwater datasets, demonstrating improved generalization of object detection models in real-world underwater applications.RS Popov et al. [18]Application of MS-based metabolomic approaches in analysis of starfish and sea cucumber bioactive compounds

Yan et al. (2024) introduced a self-guided deep learning technique for MRI image noise reduction [19]. Weimin et al. (2024) proposed enhancing liver segmentation using a deep learning approach with EAS feature extraction and multi-scale fusion [20]. Dai et al. (2023) addressed unintended bias in toxicity detection using an LSTM and attention-based approach [21]. Li et al. (2024) investigated the application of semantic networks in disease diagnosis prompts based on medical corpus [22]. Yan et al. (2024) focused on survival prediction across diverse cancer types using neural networks [23]. Xiao et al. (2024) employed convolutional neural networks for the classification of cancer cytopathology images, specifically focusing on breast cancer [24]. Wang et al. (2024) proposed a breast cancer image classification method based on deep transfer learning [25]. Li et al. (2023) explored creating accessibility linked data based on publicly available datasets [26]. These studies collectively highlight the significant impact of deep learning and neural networks across various domains of medical and computational research.

Recent advancements in machine learning and deep learning have propelled innovative research across various domains. Shen et al. (2024) utilized XGBoost for robust biomarker selection of Obsessive-Compulsive Disorder (OCD) from Adolescent Brain Cognitive Development (ABCD) data [27], highlighting significant strides in healthcare analytics. Zhang et al. (2024) applied a CNN-LSTM approach to sentiment analysis of Indian General Elections [28], showcasing the application of AI in social analysis. Wang et al. (2024) enhanced network intrusion detection using TabTransformer techniques [29], addressing critical cybersecurity challenges [30]. Additionally, Feng et al. (2024) improved heart attack prediction with eXtreme Gradient Boosting [31], while Zhao et al. (2024) leveraged BERTFusionDNN to enhance E-commerce recommendations from customer reviews [32]. These studies collectively demonstrate the breadth and depth of applications enabled by advanced machine learning methodologies.

Zhu et al. (2024) conducted a comprehensive review of knowledge distillation methods, highlighting their applications and future directions [33]. Li et al. (2024) achieved high-precision neuronal segmentation using an ensemble of YOLOX, Mask R-CNN, and UPerNet models [34], demonstrating significant progress in medical imaging analysis. Luo et al. (2024) enhanced E-commerce chatbots by integrating Falcon-7B and 16-bit full quantization techniques [35], illustrating advancements in customer service automation. Ding et al. (2024) researched optimizing lightweight small models through generating training data with ChatGPT [36], contributing to efficient AI model development. Additionally, Bao et al. (2020) presented an accurate model for predicting contextual word similarity effects based on BERT at SemEval-2020 [37], emphasizing the application of natural language processing techniques. Popokh et al. (2021) developed IllumiCore, an optimization framework for efficient Virtual Network Function (VNF) placement [38], showcasing advancements in network optimization. These studies collectively underscore the broad impact of machine learning and deep learning across healthcare, E-commerce, natural language processing, and network engineering domains.

Peng et al. (2024) proposed a Dual-Augmentor framework for domain generalization in 3D human pose estimation, showcasing advancements in computer vision applications [39]. Yin et al. (2024) utilized deep learning techniques to classify crystal systems in lithium-ion batteries, demonstrating its applicability in materials science [40]. Xie et al. (2024) developed a Conv1D-based approach for advancing legal citation text classification [41], contributing to the field of natural language processing. Furthermore [42], Su et al. (2023) introduced EdgeGYM, a reinforcement learning environment for constraint-aware NFV resource allocation [43], addressing challenges in cybersecurity infrastructure optimization. Finally, Su et al. (2024) conducted a systematic literature review on large language models for forecasting and anomaly detection [44], highlighting their role in predictive analytics and anomaly detection applications. These studies underscore the diverse applications and advancements enabled by machine learning and deep learning methodologies across various domains. Recent research demonstrates the significant benefits of utilizing deep learning and graph neural networks in the realms of software development and football formation recommendation. Li et al. (2024) explore how deep learning can optimize software development processes, leading to increased efficiency and reduced error rates. Meanwhile, Wang et al. (2024) have developed a graph neural network-based system for recommending football formations. This system can suggest the optimal formation based on real-time match data, enhancing tactical flexibility and increasing the likelihood of winning. These studies highlight the broad applications of artificial intelligence across diverse fields, driving technological advancements in their respective areas.

3.MODEL ARCHITECTURE AND METHODOLOGY

In this section, we detail the YOLOv5 model architecture and the various enhancements and tricks applied to improve its performance in detecting crown-of-thorns starfish (COTS) in underwater images. YOLOv5 is known for its balance between speed and accuracy, making it particularly suitable for real-time object detection tasks. The model architecture is divided into three main components: the backbone, the neck, and the head. Each component plays a crucial role in feature extraction, feature aggregation, and object detection, respectively.

3.1 YOLOv5 Overview

YOLOv5 (You Only Look Once version 5) is a single-stage object detection model that performs object localization and classification simultaneously. It processes the entire image in one forward pass, making it efficient for real-time applications. The YOLOv5 model used in this study is the yolov5l6 variant, which is specifically tailored for detecting small objects like COTS by increasing the resolution and depth of the model.

The general architecture of YOLOv5 is summarized by the following key components:

- **Backbone:** Extracts feature maps from the input image.
- **Neck:** Aggregates features at different scales.
- **Head:** Predicts bounding boxes and class probabilities.

The YOLOv5 model can be mathematically described as a function f that maps an input image I to a set of bounding

boxes B and corresponding class probabilities P:

$$(\boldsymbol{B}_i, \boldsymbol{P}_i)\boldsymbol{i} = \mathbf{1}^{\boldsymbol{N}} = \boldsymbol{f}(\boldsymbol{I}) \quad (\mathbf{1})$$

where N is the number of detected objects, B_i represents the coordinates of the i-th bounding box, and P_i is the class probability vector for the i-th object.

3.2 Backbone

The backbone of YOLOv5 is responsible for extracting hierarchical feature maps from the input image. It is based on the Cross Stage Partial Networks (CSPNet) architecture, which improves computational efficiency by reducing the number of gradient updates needed during training. The backbone consists of a series of convolutional layers and CSP modules that progressively downsample the image and extract increasingly abstract features.

The backbone can be described by the following stages:

1) Initial Convolution: The input image I is first passed through a convolutional layer to extract low-level features

$$F_0 = Conv(I, W_0, b_o) \quad (2)$$

where F_0 is the output feature map,W_0 and b_0 are the weights and biases of the convolutional layer.

2) CSP Blocks: The CSP blocks split the feature map into two parts, processing one part through a series of convolutional layers while retaining the other part as a shortcut connection.
Each CSP block can be represented as:

$$\boldsymbol{F}_{\boldsymbol{i}} = \boldsymbol{CSP}(\boldsymbol{F}_{\boldsymbol{i}-\mathbf{1}}) \quad (\mathbf{3})$$

where Fi denotes the feature map after the i-th CSP block.

3) Downsampling: Downsampling is performed using con- volutional layers with a stride of 2 to reduce the spatial dimensions of the feature maps while increasing the depth.

This can be mathematically expressed as

$$\boldsymbol{F}_{\boldsymbol{i}+\mathbf{1}} = \boldsymbol{Conv}(\boldsymbol{F}_{\boldsymbol{i}}, \boldsymbol{W}_{\boldsymbol{i}+\mathbf{1}}, \boldsymbol{b}_{\boldsymbol{i}+\mathbf{1}}, \boldsymbol{stride} = \mathbf{2}) \quad (\mathbf{4})$$

3.2 Neck

The neck of the YOLOv5 model aggregates feature maps from different stages of the backbone to create a feature pyramid. This helps the model to detect objects at multiple scales, which is crucial for identifying small objects like COTS. The neck uses a structure called the Path Aggregation Network (PAN), which enhances the model’s ability to capture features at different resolutions.

Key components of the neck include:

1) Feature Pyramid Network (FPN): The FPN combines features from different scales using upsampling and lateral connections. Each upsampling operation doubles the spatial resolution of the feature map:

$$\boldsymbol{F_{up}^{i} = Upsample\left(F_{high}^{i+1}, scale = 2\right)} \quad \textbf{(5)}$$

where F_{high}^{i+1} is the higher-resolution feature map from the next stage.

2) Lateral Connections: Lateral connections merge the upsampled feature map with the feature map from the cor-responding scale in the backbone:

$$\boldsymbol{F_{merged}^{i} = F_{up}^{i} + F_{low}^{i}} \quad \textbf{(6)}$$

whereF_{low}^{i} is the lower-resolution feature map from the current stage in the backbone.

3) PAN Layers: The PAN layers further process the ag-gregated features to refine them for detection. These layers are composed of convolutions and non-linear activations to enhance the feature representation:

$$\boldsymbol{F_{fina}^{i} = PAN\left(F_{merged}^{i}\right)} \quad \textbf{(7)}$$

3.2 Head

The head of the YOLOv5 model is responsible for predicting bounding boxes and class probabilities for the detected objects. It consists of several convolutional layers that output predictions at different scales, corresponding to different feature maps from the neck.

Key components of the head include:

1) Prediction Layers: The prediction layers apply con-volutions to the aggregated feature maps to generate object detection outputs. Each prediction layer outputs three values for each anchor box: the coordinates of the bounding box B, the objectness score O, and the class probabilities P:

$$O, B, P = Conv\left(F_{final}, W_{pred}, b_{pred}\right) \quad (8)$$

2) Bounding Box Regression: The bounding box regression is performed by predicting the center coordinates (x, y), width w, and height h of each bounding box relative to the anchor boxes:

$$\boldsymbol{B = \left(\sigma(t_x) + c_x, \sigma(t_x) + c_y, p_\omega e^{t_\omega}, p_h e^{t_h}\right)} \quad \textbf{(9)}$$

wheret_x, t_y, t_w,t_h are the predicted offsets, cx, cy are the center coordinates of the anchor box, and pw, ph are the anchor box dimensions.

3) Objectness Score: The objectness score O represents the likelihood that an object exists within the bounding box. It is computed using a sigmoid function to map the output to a probability between 0 and 1:

$$\boldsymbol{O = \sigma(o)} \quad \textbf{(10)}$$

where o is the raw output of the prediction layer.

4) Class Probability: The class probabilities P for each object are predicted using a softmax function to ensure that the sum of probabilities across all classes is 1:

$$P_c = \frac{ep_c}{\sum_{c'} e^{p} c'} \quad (11)$$

where pc is the raw class score for class c.

3.4 Model Enhancements and Tricks

Several enhancements and tricks were applied to the YOLOv5 model to optimize its performance for detecting COTS:

• **Increased Feature Map Sizes:** The size of the feature maps was increased by modifying the architecture to capture finer details of small objects. This involved ad-justing the convolutional layer parameters to increase the resolution of the feature maps.

• **Data Augmentation:** Extensive data augmentation techniques, including flipping, rotation, and color adjustments, were applied to increase the diversity of the training data and prevent overfitting.

• **Lower NMS Threshold:** The Non-Maximum Suppression (NMS) confidence threshold was lowered to 0.15, allowing the model to retain more potential detections for further processing.

• **Higher IoU Threshold:** The IoU threshold was raised to 0.3 to ensure that the bounding boxes are more precise, reducing the number of false positives and improving detection accuracy.

• **Integration of Tracker:** The Norfair tracking library was integrated during the inference phase to track detected objects across consecutive frames, enhancing temporal consistency and reducing false negatives.

• **Test-Time Augmentation (TTA):** Various test-time augmentation techniques were explored to improve model performance, although not all resulted in significant gains.

3.5 Dataset

The dataset used for this study was provided by the Kaggle "TensorFlow - Help Protect the Great Barrier Reef" competition. It comprises underwater images aimed at detecting crown-of-thorns starfish (COTS). The dataset includes 23,000 training images and approximately 13,000 test images, each annotated with bounding boxes indicating the presence of COTS.

1) Data Preprocessing: To prepare the dataset for training, several preprocessing steps were performed to ensure consistency and enhance model performance:

• **Image Resizing:** All images were resized to a uniform dimension to standardize the input size for the YOLOv5 model. This facilitates efficient processing and feature extraction.

• **Normalization:** Pixel values were normalized to the range [0, 1] to improve convergence during model training. This step helps in mitigating the effect of varying lighting conditions in underwater images.

• **Data Augmentations:** Extensive data augmentation techniques were applied to increase dataset diversity and improve model robustness. These techniques included random flipping, rotation, and color adjustments, which helped in generalizing the model to different underwater scenarios.

2) Special Tricks: Several specific tricks were applied to the dataset to further enhance the model's performance:

• **Selective Augmentation:** Data augmentation was selectively applied to images containing COTS to focus the model on learning diverse representations of the target object.

• **Hard Negative Mining:** Images without COTS were carefully selected to train the model to accurately distinguish between backgrounds and potential false positives, thereby reducing the number of false detections.

• **Sequential Data Handling:** Given that the images were extracted from video footage, maintaining the sequence order helped in utilizing temporal information for improved detection and tracking of COTS across frames.

By employing these preprocessing steps and tricks, the dataset was optimized to train a robust YOLOv5 model capable of accurately detecting COTS in diverse and challenging underwater environments.

3.6 Loss Function

The loss function in YOLOv5 is crucial for training the model to accurately detect COTS by optimizing the predictions of bounding boxes and class probabilities. It consists of three main components: bounding box regression loss, objectness loss, and class probability loss.

1) Bounding Box Regression Loss: YOLOv5 uses the Generalized Intersection over Union (GIoU) loss to measure the accuracy of the predicted bounding box relative to the ground truth. The GIoU loss is defined as:

$$\mathcal{L}_{GIou} = 1 - \frac{A \cap B}{A \cup B} + \frac{|C - (A \cup B)|}{|C|} \tag{12}$$

where A and B are the predicted and ground-truth boxes, respectively, and C is the smallest enclosing box.

2) Objectness Loss: The objectness loss uses binary cross-entropy to assess whether a predicted box contains an object. It is calculated as:

$$\mathcal{L}_{obj} = -\left(y \log(\widehat{y}) + (1 - y) \log(1 - \hat{y})\right) \tag{13}$$

where y is the ground-truth objectness score, and ˆy is the predicted score.

3) Class Probability Loss: The class probability loss evaluates the accuracy of class predictions using multi-class cross-entropy:

$$\mathcal{L}_{cls} = -\sum_{c=1}^{C} y_c \log(\hat{y}_c) \tag{14}$$

where y_c is the ground-truth probability for class c, and $\hat{y}_c$ is the predicted probability

4) Overall Loss Function: The total loss is a weighted sum of these components:

$$\boldsymbol{\mathcal{L}} = \lambda_{\mathbf{GIou}} \boldsymbol{\mathcal{L}}_{\mathbf{GIou}} + \lambda_{\mathbf{ogj}} \boldsymbol{\mathcal{L}}_{\mathbf{obj}} + \lambda_{\mathbf{cls}} \boldsymbol{\mathcal{L}}_{\mathbf{cls}} \tag{15}$$

whereλ_{GIou}, λ_{obj}, and λ_{cls} are the respective weights.

5) Tricks for Enhanced Performance:

• **Dynamic Loss Scaling:** Adjusts weights dynamically to balance different loss components.
• **Focal Loss for Objectness:** Mitigates class imbalance by focusing on harder examples.
• **Label Smoothing:** Reduces overconfidence in predictions, improving generalization.

3.7 Evaluation Metric

We use F2 score as the evaluation metric, which places greater emphasis on recall to ensure fewer crown-of-thorns starfish (COTS) are missed, even at the cost of some false positives. The F2 score is calculated as:

$$F2 = \frac{5 * Precision * Recall}{4 * Precision + Recall} \tag{16}$$

The evaluation involves sweeping across multiple intersection over union (IoU) thresholds from 0.3 to 0.8, with a step size of 0.05. At each threshold, the F2 score is calculated, and the final score is the mean of these individual F2 scores. The IoU for each bounding box is computed as:

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{17}$$

A predicted bounding box is considered a true positive if its IoU with a ground truth box exceeds the current threshold. Unmatched predicted boxes are counted as false positives, and unmatched ground truth boxes are counted as false negatives. The F2 scores are averaged across all IoU thresholds to produce the final evaluation score.

3.8 Experiment Results

In this section, we present the results of various configurations and enhancements applied to the YOLOv5 model for detecting crown-of-thorns starfish (COTS). Table t summarizes Table 1 he public scores for each configuration and enhancement.

The results in Table Table 1 clearly demonstrate the effectiveness of the incremental enhancements applied to the YOLOv5 model. The final score of 0.715 underscores the cumulative impact of these enhancements, validating their effectiveness in improving the model's accuracy and robustness in detecting COTS in challenging underwater environments.

TABLE I: Performance of YOLOv5 Configurations

Configuration and Enhancement	Score
YOLOv5 Baseline	0.630
Data Augmentation (Enhanced in YAML)	0.634
Lower NMS Confidence Threshold to 0.15	0.657
Increase IoU Threshold to 0.3	0.670
Test-Time Augmentation (TTA) Flip	No Gain
Image Enhancement (e.g., CLAHE)	No Gain
Integration of Video Tracker	0.690
WBF Fusion of Tracker and Original Targets	0.712
Increase Image Size to 3100 and More Epochs	0.715

4. CONCLUSION

In conclusion, the incremental enhancements applied to the YOLOv5 model have significantly bolstered its ability to accurately detect crown-of-thorns starfish (COTS) in complex underwater environments. The final model, achieving a score of 0.715, demonstrates a robust and precise detection capability, essential for real-time monitoring and conservation efforts. The use of data augmentation, threshold adjustments, and advanced tracking integration has proven effective in addressing the challenges of underwater object detection. These advancements highlight the potential of deep learning techniques in environmental monitoring, offering a powerful tool for safeguarding coral reef ecosystems against invasive species like COTS.

REFERENCES

[1] Fayaz, S., Parah, S. A., & Qureshi, G. J. (2022). Underwater object detection: architectures and algorithms–a comprehensive review. Multimedia Tools and Applications, 81(15), 20871-20916.

[2] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

[3] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

[4] Lu, Y., & Lu, G. (2021). Superthermal: Matching thermal as visible through thermal feature exploration. IEEE Robotics and Automation Letters, 6(2), 2690-2697.

[5] Rekavandi, A. M., Xu, L., Boussaid, F., Seghouane, A. K., Hoefs, S., & Bennamoun, M. (2022). A guide to image and video based small object detection using deep learning: Case study of maritime surveillance. arXiv preprint arXiv:2207.12926.

[6] Liang, X., Zhang, J., Zhuo, L., Li, Y., & Tian, Q. (2019). Small object detection in unmanned aerial vehicle images using feature fusion and scaling-based single shot detector with spatial context analysis. IEEE Transactions on Circuits and Systems for Video Technology, 30(6), 1758-1770.

[7] Lin, Z., & Xu, F. (2023, July). Simulation of Robot Automatic Control Model Based on Artificial Intelligence Algorithm. In 2023 2nd International Conference on Artificial Intelligence and Autonomous Robot Systems (AIARS) (pp. 535-539). IEEE.

[8] Wang, Q., Du, Z., Jiang, G., Cui, M., Li, D., Liu, C., & Li, W. (2022). A Real-Time Individual Identification Method for Swimming Fish Based on Improved Yolov5. Available at SSRN 4044575.

[9] Chen, Z., Du, M., Yang, X. D., Chen, W., Li, Y. S., Qian, C., & Yu, H. Q. (2023). Deep-learning-based automated tracking and counting of living plankton in natural aquatic environments. Environmental Science & Technology, 57(46), 18048-18057.
[10] Chen, H., Yang, Y., & Shao, C. (2021). Multi-task learning for data-efficient spatiotemporal modeling of tool surface progression in ultrasonic metal welding. Journal of Manufacturing Systems, 58, 306-315.
[11] Alawode, B., Guo, Y., Ummar, M., Werghi, N., Dias, J., Mian, A., & Javed, S. (2022). Utb180: A high-quality benchmark for underwater tracking. In Proceedings of the Asian Conference on Computer Vision (pp. 3326-3342).
[12] Lu, Y., & Lu, G. (2021). An alternative of lidar in nighttime: Unsupervised depth estimation based on single thermal image. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 3833-3843).
[13] Xie, L., Wang, H., Wang, Z., & Cheng, L. (2020, July). DHD-Net: A novel deep-learning-based dehazing network. In 2020 International Joint Conference on Neural Networks (IJCNN) (pp. 1-7). IEEE.
[14] Liu, K., Peng, L., & Tang, S. (2023). Underwater object detection using TC-YOLO with attention mechanisms. Sensors, 23(5), 2567.
[15] Peng, Q., Ding, Z., Lyu, L., Sun, L., & Chen, C. (2022). RAIN: regularization on input and network for black-box domain adaptation. arXiv preprint arXiv:2208.10531.
[16] Tian, G., & Xu, Y. (2022). A Study on the Typeface Design method of Han Characters imitated Tangut. Advances in Education, Humanities and Social Science Research, 1(2), 270-270.
[17] Song, S., Li, Y., Huang, Q., & Li, G. (2021). A new real-time detection and tracking method in videos for small target traffic signs. Applied Sciences, 11(7), 3061.
[18] Popov, R. S., Ivanchina, N. V., & Dmitrenok, P. S. (2022). Application of MS-based metabolomic approaches in analysis of starfish and sea cucumber bioactive compounds. Marine Drugs, 20(5), 320.
[19] Yan, X., Xiao, M., Wang, W., Li, Y., & Zhang, F. (2024). A Self-Guided Deep Learning Technique for MRI Image Noise Reduction. Journal of Theory and Practice of Engineering Science, 4(01), 109-117.
[20] Weimin, W. A. N. G., Yufeng, L. I., Xu, Y. A. N., Mingxuan, X. I. A. O., & Min, G. A. O. (2024). Enhancing Liver Segmentation: A Deep Learning Approach with EAS Feature Extraction and Multi-Scale Fusion. International Journal of Innovative Research in Computer Science & Technology, 12(1), 26-34.
[21] Dai, W., Tao, J., Yan, X., Feng, Z., & Chen, J. (2023, November). Addressing Unintended Bias in Toxicity Detection: An LSTM and Attention-Based Approach. In 2023 5th International Conference on Artificial Intelligence and Computer Applications (ICAICA) (pp. 375-379). IEEE.
[22] Li, Y., Wang, W., Yan, X., Gao, M., & Xiao, M. (2024). Research on the application of semantic network in disease diagnosis prompts based on medical corpus. International Journal of Innovative Research in Computer Science & Technology, 12(2), 1-9.
[23] Yan, X., Wang, W., Xiao, M., Li, Y., & Gao, M. (2024). Survival prediction across diverse cancer types using neural networks. arXiv preprint arXiv:2404.08713.
[24] Xiao, M., Li, Y., Yan, X., Gao, M., & Wang, W. (2024). Convolutional neural network classification of cancer cytopathology images: taking breast cancer as an example. arXiv preprint arXiv:2404.08279.
[25] Wang, W., Gao, M., Xiao, M., Yan, X., & Li, Y. (2024). Breast Cancer Image Classification Method Based on Deep Transfer Learning. arXiv preprint arXiv:2404.09226.
[26] Li, Y., Yan, X., Xiao, M., Wang, W., & Zhang, F. (2023, December). Investigation of creating accessibility linked data based on publicly available accessibility datasets. In Proceedings of the 2023 13th International Conference on Communication and Network Security (pp. 77-81).
[27] Shen, X., Zhang, Q., Zheng, H., & Qi, W. (2024). Harnessing XGBoost for Robust Biomarker Selection of Obsessive-Compulsive Disorder (OCD) from Adolescent Brain Cognitive Development (ABCD) data. ResearchGate, May.
[28] Zhang, N., Xiong, J., Zhao, Z., Feng, M., Wang, X., Qiao, Y., & Jiang, C. (2024). Dose My Opinion Count? A CNN-LSTM Approach for Sentiment Analysis of Indian General Elections. Journal of Theory and Practice of Engineering Science, 4(05), 40-50.
[29] Wang, X., Qiao, Y., Xiong, J., Zhao, Z., Zhang, N., Feng, M., & Jiang, C. (2024). Advanced network intrusion detection with tabtransformer. Journal of Theory and Practice of Engineering Science, 4(03), 191-198.
[30] Su, J., Nair, S., & Popokh, L. (2022, November). Optimal resource allocation in sdn/nfv-enabled networks via deep reinforcement learning. In 2022 IEEE Ninth International Conference on Communications and Networking (ComNet) (pp. 1-7). IEEE.
[31] Feng, M., Wang, X., Zhao, Z., Jiang, C., Xiong, J., & Zhang, N. (2024). Enhanced Heart Attack Prediction Using eXtreme Gradient Boosting. Journal of Theory and Practice of Engineering Science, 4(04), 9-16.

[32] Zhao, Z., Zhang, N., Xiong, J., Feng, M., Jiang, C., & Wang, X. (2024). Enhancing E-commerce Recommendations: Unveiling Insights from Customer Reviews with BERTFusionDNN. Journal of Theory and Practice of Engineering Science, 4(02), 38-44.
[33] Zhu, E. Y., Zhao, C., Yang, H., Li, J., Wu, Y., & Ding, R. (2024). A Comprehensive Review of Knowledge Distillation-Methods, Applications, and Future Directions. International Journal of Innovative Research in Computer Science & Technology, 12(3), 106-112.
[34] Li, Z., Yin, Y., Wei, Z., Luo, Y., Xu, G., & Xie, Y. (2024). High-Precision Neuronal Segmentation: An Ensemble of YOLOX, Mask R-CNN, and UPerNet. Journal of Theory and Practice of Engineering Science, 4(04), 45-52.
[35] Luo, Y., Wei, Z., Xu, G., Li, Z., Xie, Y., & Yin, Y. (2024). Enhancing E-commerce Chatbots with Falcon-7B and 16-bit Full Quantization. Journal of Theory and Practice of Engineering Science, 4(02), 52-57.
[36] Ding, R., Zhu, E. Y., Zhao, C., Yang, H., Li, J., & Wu, Y. (2024). Research on Optimizing Lightweight Small Models Based on Generating Training Data with ChatGPT. Journal of Industrial Engineering and Applied Science, 2(2), 39-45.
[37] Bao, W., Che, H., & Zhang, J. (2020, December). Will_Go at SemEval-2020 Task 3: An accurate model for predicting the (graded) effect of context in word similarity based on BERT. In Proceedings of the Fourteenth Workshop on Semantic Evaluation (pp. 301-306).
[38] Popokh, L., Su, J., Nair, S., & Olinick, E. (2021, September). IllumiCore: Optimization Modeling and Implementation for Efficient VNF Placement. In 2021 International Conference on Software, Telecommunications and Computer Networks (SoftCOM) (pp. 1-7). IEEE.
[39] Peng, Q., Zheng, C., & Chen, C. (2024). A Dual-Augmentor Framework for Domain Generalization in 3D Human Pose Estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 2240-2249).
[40] Yin, Y., Xu, G., Xie, Y., Luo, Y., Wei, Z., & Li, Z. (2024). Utilizing Deep Learning for Crystal System Classification in Lithium-Ion Batteries. Journal of Theory and Practice of Engineering Science, 4(03), 199-206.
[41] Xie, Y., Li, Z., Yin, Y., Wei, Z., Xu, G., & Luo, Y. (2024). Advancing Legal Citation Text Classification A Conv1D-Based Approach for Multi-Class Classification. Journal of Theory and Practice of Engineering Science, 4(02), 15-22.
[42] Peng, Q., Zheng, C., & Chen, C. (2023). Source-free domain adaptive human pose estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 4826-4836).
[43] Su, J., Nair, S., & Popokh, L. (2023, February). EdgeGYM: a reinforcement learning environment for constraint-aware NFV resource allocation. In 2023 IEEE 2nd International Conference on AI in Cybersecurity (ICAIC) (pp. 1-7). IEEE.
[44] Su, J., Jiang, C., Jin, X., Qiao, Y., Xiao, T., Ma, H., ... & Lin, J. (2024). Large Language Models for Forecasting and Anomaly Detection: A Systematic Literature Review. arXiv preprint arXiv:2402.10350.
[45] Li, K., Zhu, A., Zhou, W., Zhao, P., Song, J., & Liu, J. (2024). Utilizing deep learning to optimize software development processes. arXiv preprint arXiv:2404.13630.
[46] Zeyu Wang, Yue Zhu, Zichao Li, Zhuoyue Wang, Hao Qin, and Xinqi Liu. “Graph Neural Network Recommendation System for Football Formation”. Applied Science and Biotechnology Journal for Advanced Research, vol. 3, no. 3, May 2024, pp. 33-39, doi:10.5281/zenodo.12198843.