CLAM MEAT DETECTION ALGORITHM BASED ON IMPROVED YOLOv5s
/
基于改进 YOLOv5s 的蛤肉检测算法

Xinkai JIAO, Xiangcai ZHANG[*)], Zhongcai WEI, Xianliang WANG, Xiupei CHENG, Pingchuan MA
Shandong University of Technology, Institute of Modern Agricultural Equipment, ZiBo (255000), China
Tel: +86-15169235925; E-mail: zxcai0216@163.com
DOI: https://doi.org/10.35633/inmateh-76-52

Keywords: *machine vision; Ruditapes philippinarum; targeted detection; YOLO*

ABSTRACT
Intelligent and accurate shelling technology is essential for improving the quality of clam products. To enable the rapid and precise localization of clam meat in Ruditapes philippinarum (with half-shell) on an automated processing line, an improved clam meat detection algorithm - EET-YOLOv5, based on YOLOv5s - is proposed. This algorithm enables real-time detection and localization of clam meat on the production line. It integrates the Efficient Local Attention (ELA) mechanism to enhance target localization, adopts the EIoU loss function to reduce bounding box regression error, and replaces the original detection head with a TSCODE decoupled head to improve detection accuracy. The algorithm achieved a Precision of 93.03%, Recall of 97.03%, and mean Average Precision (mAP) of 93.55%, with a detection speed of 13.3 ms. Compared to YOLOv4, Faster R-CNN, SSD, and the standard YOLOv5 series, EET-YOLOv5 demonstrated superior performance. It was deployed on a test workbench for positioning experiments, achieving an average response time of 1.8 seconds and a positioning success rate of 92.7%, indicating its suitability for automated clam shell-meat separation production lines.

摘要
智能化、精确化的脱壳技术是提高蛤肉产品质量的关键。为了实现自动化加工生产线上对开半壳的菲律宾蛤仔中蛤肉的快速准确识别，提出了一种基于 YOLOv5s 改进而来的蛤肉检测算法（EET-YOLOv5），可以对生产线上的蛤肉进行实时定位识别。该算法融合 Efficient Local Attention（ELA）注意力机制，能够有效捕捉目标位置；采用 EIOU 损失函数，减少边界框回归损失；使用 TSCODE 解耦头替换原有检测头，提高检测准确率。该算法检测蛤肉的精确率、召回率和平均精度均值分别达到 93.03%，97.03%，93.55%，检测速度达到 13.3ms。将其与 YOLOv4、Faster-RCNN、SSD 和 YOLOv5 系列等算法比较具有明显优势。将其部署在实验台上进行定位试验，平均耗时 1.8 秒，成功率 92.7%，适用于自动化蛤仔壳肉分离生产线。

INTRODUCTION
As a bivalve shellfish with low farming cost and short cycle time, Philippine clams (*Ruditapes philippinarum*) are widely farmed worldwide for their high food and medicinal values (*Chang et al., 2007*). Global production exceeds 3 million tons annually (*Lin et al., 2022*), and this large volume has driven continuous advancements in its processing technology. Shell-flesh separation is an important part of Ruditapes philippinarum processing (*Zhao et al., 2023*). Manual shelling is time-consuming and laborious, so automatic shelling technology has emerged. The traditional automatic shell-meat separation method is mainly based on thermal shelling and mechanical shelling. Thermal shelling involves heating shellfish at high temperatures to inactivate or denature the adductor muscle fibers, thereby enabling the separation of the meat from the shell. In contrast, mechanical shelling uses mechanical methods to open the shell and extract the meat without the need for heat treatment, resulting in a fresher product. However, most current mechanical shelling methods suffer from limitations such as low accuracy and complex shell-clamping mechanisms, which hinder production efficiency. Additionally, thermal shelling significantly compromises the freshness of the shellfish, making it unsuitable for meeting market demands (*Zhang et al., 2013)*.

With the development of intelligent technology, the target detection network plays a key role in shellfish processing. Wang Haifeng proposed a new shellfish recognition algorithm, which uses Gabor filter combined with 2D PCA to extract shellfish image features. The average correct recognition rate of this method on the self-made data set reaches 96.7 % (*Wang et al., 2016*). *Feng Yiran et al., (2022)*, developed a shellfish recognition algorithm based on Faster R-CNN. DenseNet was used instead of the feature extraction module to fuse features at different levels, and the NMS algorithm was optimized to overcome the problems of shellfish overlap and omission, and the shellfish classification accuracy was improved by 4 %.

Zhang Yang et al., (2022), proposed the FCnet algorithm for accurate detection and recognition of multi-target shellfish. He added the AttResNet101 backbone network containing the multi-spectral channel attention mechanism to the DeFCN network to obtain rich information of the original features, and improved the feature extraction module to improve the accuracy of small target recognition. *Dong Zhaopeng et al., (2023),* used the CST-YOLOv5 algorithm to identify mussels, and the mAP index was improved by 1.583% compared with the original algorithm. *Yu Zhe et al., (2025),* developed a freshwater snail detection algorithm based on YOLOv8-OBB, which effectively improved the detection accuracy for small targets, and the mAP0.5 reached 80.6 %. *Liu Zhenlong et al., (2025),* used the YOLOv10-MECAS algorithm to identify sea cucumbers, effectively overcoming the problem of low underwater visibility. On their custom dataset, the algorithm achieved a mAP@0.5 of 90.4%.

Intelligent processing is the key technology to improve the production efficiency and product quality of shellfish. For the Ruditapes philippinarum, the clams that have been opened on the production line can be dynamically monitored in real time to accurately locate the clam meat and guide the clamping device to take it out, thereby improving the success rate of shell meat separation and ensuring the freshness of the clam meat. Therefore, a target detection network based on YOLOv5s is designed to detect clam meat on the production line and provide accurate positioning for the intelligent clamping device.

MATERIALS AND METHODS

Image acquisition

Shelled clams (Ruditapes philippinarum) were photographed, and the captured images were processed and used for training the network model. The images were taken in the Marine Products Processing Laboratory of Shandong University of Technology. The clams used in this study had naturally grown for 16 months after sowing. To replicate conditions on a processing line, the clams were placed naturally on a small conveyor belt with the shell openings facing upward. The clams were neither fixed in position nor stacked. To ensure diversity in the dataset, images were captured from various angles, distances, and under different lighting conditions. A smartphone was used as the imaging device. A total of 857 images were taken at a resolution of 4096 × 3072 pixels, from which 775 images were selected after screening. A sample of the collected clam meat images is shown in Figure 1.

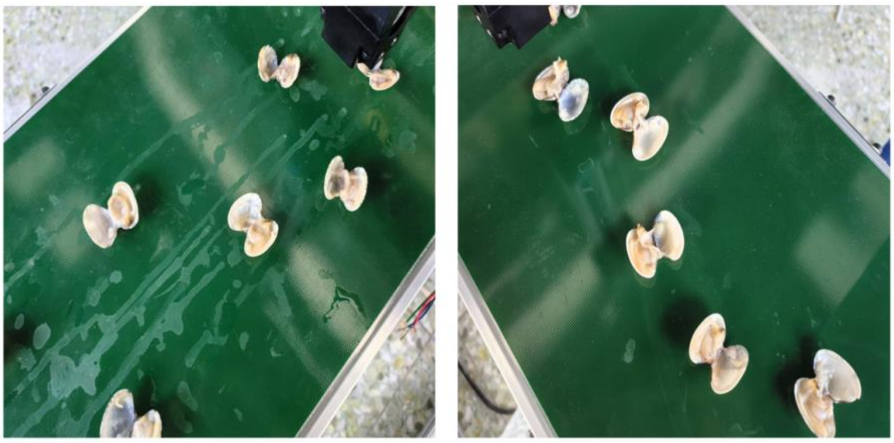


Fig. 1 - Part of the clam meat image collected

Data preprocessing and dataset construction

Before training the target detection network, the captured image data were preprocessed to construct a usable dataset. The LabelImg tool was used to annotate the location of clam meat targets by drawing bounding boxes. The corresponding location information was saved in TXT annotation files, and both the images and annotations were organized into a standard YOLO-format dataset. Since the manually captured images lacked sufficient scene complexity and quantity for effective model training, data augmentation was applied to expand and diversify the dataset. Augmentation techniques included random rotation, scaling, aspect ratio adjustment, contrast enhancement, and the addition of Gaussian noise. The final dataset consisted of 2,300 images, with 150 images designated as the test set. The remaining 2,150 images were randomly divided into a training set and a validation set at an 8:2 ratio, resulting in 1,720 training samples and 430 validation samples.

YOLOv5 algorithm

YOLOv5 is a single-stage object detection algorithm developed as an improvement over previous models in the YOLO series, particularly YOLOv4. In the input stage, YOLOv5 incorporates Mosaic data augmentation and an adaptive anchor box mechanism, which significantly enhance the detection performance, especially for small targets. The overall structure of YOLOv5 is illustrated in Figure 2.

In the backbone network, the CSP structure is employed to reduce the number of parameters. The SPPF module is integrated to improve the detection of objects at varying scales without compromising inference speed. In the neck, the FPN combined with PAN enables effective multi-scale feature fusion and target localization. With its high detection accuracy and low deployment cost, YOLOv5 is well-suited for rapid deployment on mobile and embedded devices. Therefore, it was selected as the base algorithm for target detection in this study.

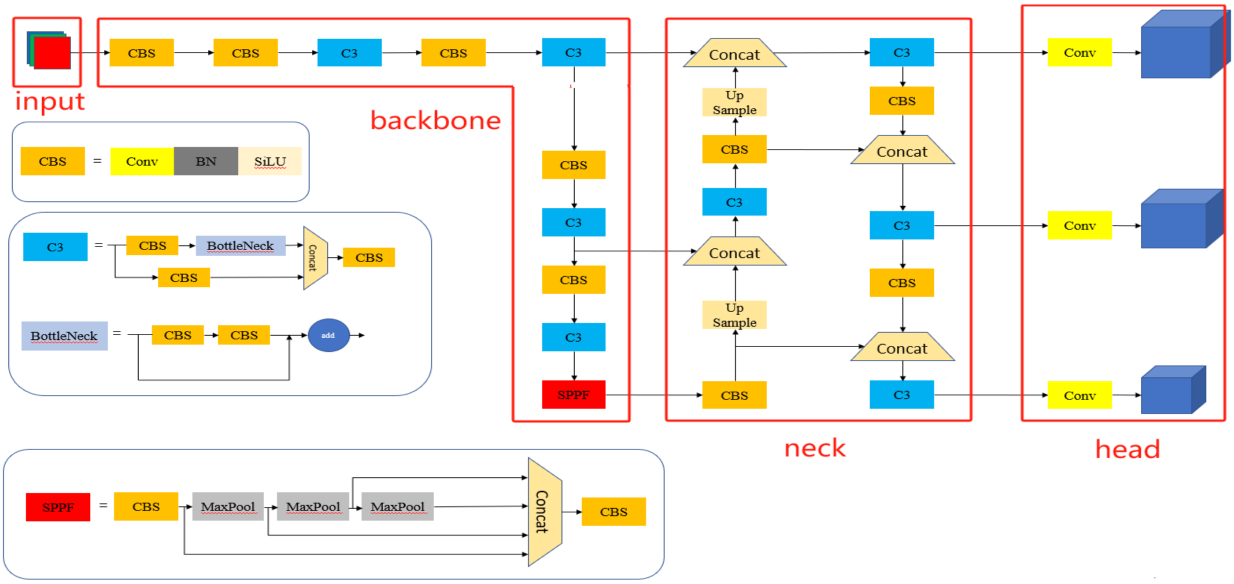


Fig. 2 - YOLOv5 network architecture diagram

Conv denotes the convolution operation; BN is normalization operation; SiLU is the Sigmoid Linear Unit activation function; Concat represents channel-wise concatenation and Add indicates element-wise addition

ELA attention mechanisms

Attention mechanisms enable networks to focus on meaningful parts of images. The attention mechanism can act on two dimensions of channel and space, namely channel attention and spatial attention. Common attention mechanisms include SEnet (*Hu et al., 2019*), ECA (*Wang et al., 2020*), CBAM (*Woo et al., 2018*), CA (*Hou et al., 2021*), ELA (*Xu et al., 2024*), etc.

In this study, the focus is on the localization of clam meat, which exhibits an irregular shape and a yellowish color after the shell is opened. Under varying lighting conditions, the clam meat often shares a similar color with the inner surface of the shell, making accurate localization challenging. To address this, it is crucial to minimize channel information loss during the extraction of spatial and channel features in order to obtain a clearer target contour. To achieve this, the Efficient Local Attention (ELA) module is introduced to enhance feature map processing and preserve critical information for more accurate contour detection.

ELA performs pooling operations along the width and height dimensions of the input tensor to extract global features in both directions. These features preserve both channel and spatial information, aiding the neural network in accurately locating target coordinates. Additionally, the pooling mechanism provides a global receptive field, enabling the establishment of long-range dependencies between elements. To further refine feature extraction while maintaining computational efficiency, one-dimensional convolution is applied. This approach ensures effective feature representation without reducing the number of channels.

GroupNorm is applied to normalize data batches (*Wu et al., 2018*), effectively mitigating the problem of vanishing gradients and enhancing the generalization capability of the network. Using the Sigmoid activation function, directional weight coefficients are computed and subsequently multiplied with the input feature map to refine the feature representation. Integrating the ELA module before the SPPF structure enables more accurate localization of regions of interest and improves the network's feature learning capacity. The architecture of this structure is illustrated in Figure 3.

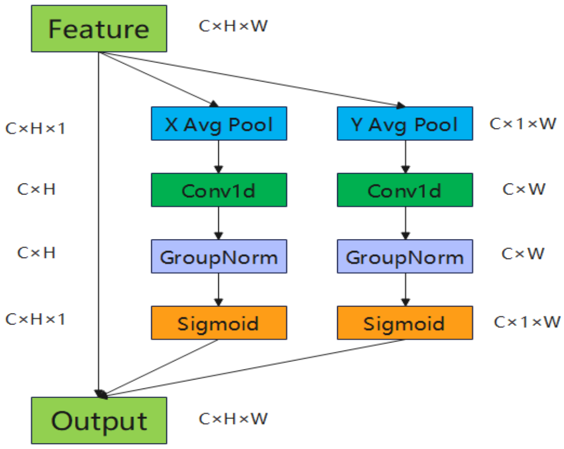


Fig. 3 - ELA attention mechanism architecture diagram.
X,Y Avg Pool is average pooling operation along the X and Y directions; Conv1d is one-dimensional convolution operation; GroupNorm is group normalization layer; Sigmoid is the sigmoid activation function

The ELA attention mechanism is calculated as follows:

$$Z_c^h(h) = \frac{1}{W}\sum_{0\le i<W} x_c(h,i) \tag{1}$$

$$Z_c^w(w) = \frac{1}{H}\sum_{0\le j<H} x_c(j,w) \tag{2}$$

$$y^h = \delta\left(G_n\left(F_h(z^h)\right)\right) \tag{3}$$

$$y^w = \delta\left(G_n\left(F_w(z^w)\right)\right) \tag{4}$$

$$Y = x_c \times y^h \times y^w \tag{5}$$

Here, $Z_c^h(h)$ is the pooled output of the c th channel at height h. W is the Width of input features. $x_c\,(i,j)$ is the elements at (i,j) in channel c. $Z_c^w(w)$ is the pooled output of the c-th channel at width w. H is the Width of input features. z^h is the characteristic map after pooling in the vertical direction. F_h is One-dimensional convolution in the vertical direction. G_n is grouping normalization. δ is sigmoid activation function. y^h is the vertical attention weights. z^w is the characteristic map after pooling in the vertical direction. F_w is One-dimensional convolution in the vertical direction. y^w is the vertical attention weights. Y is the output result.

Loss function optimization

The loss function is the core of the deep learning parameter optimization process. The model parameters are adjusted by minimizing the loss function to improve the prediction accuracy and generalization ability of the model. The loss function mainly considers the influence factors such as Euclidean distance, intersection over union and aspect ratio between the real box and the prediction box. The most widely used loss functions are GIOU (*Rezatofighi et al., 2019*), CIOU (*Zheng et al., 2020*), and SIOU (*Gevorgyan, 2022*), etc. The original network of YOLOv5 uses CIOU as the loss function. CIOU considers the overlap area, distance and aspect ratio of the predicted box and the ground truth box. However, the penalty term on the aspect ratio does not take into account the same aspect ratio of the two frames and the increase and decrease of the aspect gradient at the same time.

Therefore, this paper introduces EIOU to address this issue, which overcomes the limitations of conventional IoU-based losses by decomposing the aspect ratio penalty into two separate components - height and width - and calculating them independently. This approach enhances detection accuracy and accelerates network training. The EIoU loss is calculated as follows:

$$EIOU = IOU - \frac{\rho^2(b,b^{gt})}{(h^c)^2+(w^c)} - \frac{\rho^2(h,h^{gt})}{h^c} - \frac{\rho^2(w,w^{gt})}{w^c} \tag{6}$$

$$L_{EIOU} = 1 - EIOU \tag{7}$$

Here, IOU is the intersection over union of ground truth box and predicted box. h,h^{gt} is the height of gt-box (ground truth box) and p-box (predicted box). w,w^{gt} is the Width of gt-box and p-box. $\rho(b,b^{gt})$ is the Euclidean distance between the gt-box and p-box, and h^c, w^c is the height and width of the smallest external rectangle. L_{EIOU} is the Value of EIOU's losses.

Detection head improvement

Clam meat detection falls under the category of small target detection, with irregular and indistinct edges. Therefore, high localization accuracy is essential, and the TSCODE decoupled head is employed to enhance target detection performance. The YOLOv5 original network structure uses a coupled detection head to complete the classification and positioning tasks of the target at the same time. However, for some feature regions, there are rich semantic features inside and a lot of boundary texture on the boundary. The classification task needs to pay attention to the internal position, while the positioning task needs to pay attention to the boundary position. There is a conflict between the two requirements. If the same feature map is used, spatial misalignment may occur, affecting the positioning accuracy (*Song et al., 2020; Wu et al.,2020*). The TSCODE decoupled header enables the two to be performed separately. It uses the Semantic Context Encoding (SCE) module to generate feature encoding containing high semantic information, and Detail Preserving Encoding (DPE) to provide high-resolution feature maps containing edge information to perform the classification and localization tasks respectively. Its structure is shown in Fig. 4.

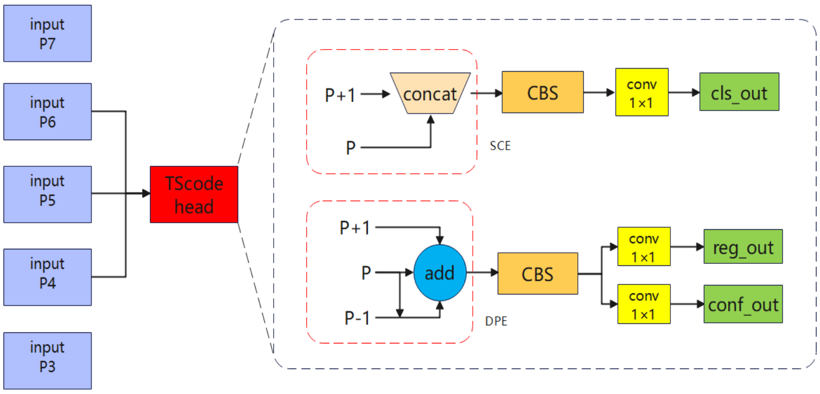


Fig. 4 - TSCODE decoupled head architecture diagram.
SCE is Semantic context encoding operation; DPE is Detail preserving encoding operation;
cls_out is the classification results; reg_out is the Predicted Bounding Box location and size information;
conf_out is the confidence for predicted bounding box

Set up the experimental platform

A test rig was constructed to verify the real-world deployment performance of the proposed algorithm. The test system consists of a small conveyor, a gantry-type test rig, an XY linear synchronous belt, a binocular camera, an Arduino control board, and a laser emitter. To accurately analyze the spatial relationship between the laser emitter and the clam meat, multiple coordinate systems were established: O_A at the position of the shell meat, O_B at the position of the laser emitter, O_C at the position of the binocular camera, and O_W at the base of the test bench. The configuration of the experimental setup and coordinate systems is illustrated in Fig. 5.

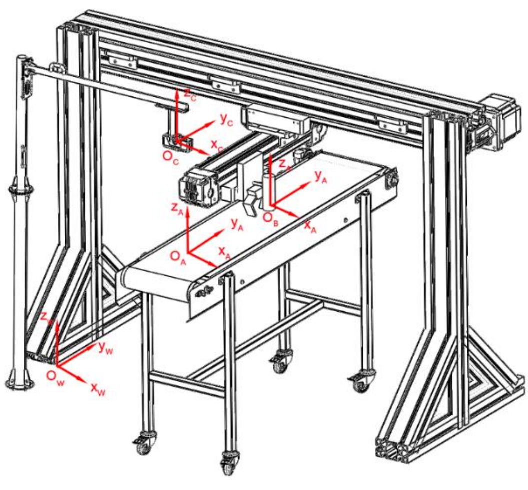


Fig. 5 - Test rig schematic diagram

RESULTS

Experimental environment

The core processor (CPU) used in this test was Intel I7-11700K, and the graphics card was NVIDIA RTX A4000 24G. The operating system was Win11, The deep learning framework was built using PyTorch 1.8.1, CUDA 11.1, and TensorFlow 2.4.1, with Python 3.8.19 as the programming language. At the beginning of training, the input image size was set to 640 * 640, the training was optimized by SGD, the maximum number of iterations was set to 1200 rounds, the batch_size was set to 8, and the maximum learning rate was set to 0.001. The cosine annealing learning rate adjustment algorithm was used to dynamically adjust the learning rate, so as to accelerate the convergence speed of the model and enhance the generalization ability of the model.

Evaluation indicators

To assess the model's real-world performance, several evaluation indicators are used, including Precision (P), Recall (R), mean Average Precision (mAP), number of model parameters, box loss (box_loss), and object loss (obj_loss). The calculation formulas are as follows:

$$P = \frac{TP}{TP + FP} \times 100\% \tag{8}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{9}$$

$$AP = \int_0^1 P(R)dR \tag{10}$$

$$mAP = \frac{\sum AP}{N} \tag{11}$$

Here, TP (True Positives) represents the number of samples that were correctly rejected, FP (False Positives) represents the number of incorrectly rejected samples. FN represents the number of samples that were missed, AP represents the average precision of the model's assessment of a class of targets and mAP represents the mean accuracy of all categories of targets.

Box_loss represents the error between the ground truth box and the predicted box.

Obj_loss represents the probability of error in the model's judgement of the presence or absence of an object in the predicted box.

Comparison of ablation experimental performance

To evaluate the impact of individual improvements on the target detection performance, ablation experiments were conducted using YOLOv5s as the baseline model. Three components were tested: the ELA attention mechanism, the SIoU loss function, and the TSCODE decoupled head. Each component was added separately to the baseline, and then all were integrated to form the final improved YOLOv5s model. The results of the ablation experiments are summarized in Table 1, where “√” indicates that the corresponding component was applied, and “—” indicates that it was not.

Table 1

Results of the ablation experiment

ELA	EIOU	TSCODE	Precision [%]	Recall [%]	mAP_0.5* [%]
—	—	—	91.98	94,46	91.66
√	—	—	91.92	94.81	91.70
—	√	—	92.16	95.78	92.48
—	—	√	92.50	96.51	92.57
√	√	√	93.03	97.03	93.55

Note: * mAP_0.5 is the mean average precision of each category when the Intersection over Union threshold is 0.5

It can be observed from Table 1 that adding the ELA attention mechanism alone results in a change of -0.06% in Precision, and slight improvements of 0.35% in Recall and 0.04% in mAP. When the SIoU loss function is applied individually, the three indicators increase by 0.18%, 1.32%, and 0.82%, respectively.

The introduction of the TSCODE decoupled head alone leads to improvements of 0.52% in Precision, 2.05% in Recall, and 0.91% in mAP.

When the modalities for improvement are combined, the accuracy rate of the improved YOLOv5s model reaches 93.03 %, the recall rate is 97.03 %, and the mAP value is 93.55 %. Compared with YOLOV5s, it increased by 1.05 %, 2.57 % and 1.89 % respectively.

The results demonstrate that the use of the TSCODE detection head, which decouples classification and localization tasks, effectively prevents mutual interference between the two and significantly enhances detection accuracy. The SIoU loss function improves the convergence behavior of the detection frame. While the ELA attention mechanism enhances Recall, it causes a slight decrease in Precision. The most notable performance improvement is achieved when all three components are combined, yielding results that outperform the original YOLOv5s algorithm. Accordingly, the improved model is named EET-YOLOv5.

Performance comparison of YOLOv5 algorithms for different sizes

To evaluate whether the improved model maintains high detection accuracy while remaining lightweight, a performance comparison was conducted against more complex models from the YOLOv5 series. All models were tested under identical conditions. The comparison results are presented in Table 2.

Table 2

Comparative experimental results

Model	Precision	Recall	mAP_0.5	Model parameter quantity	Detection speed
	[%]	[%]	[%]	[M]	[ms]
EET-YOLOv5	93.03	97.03	93.55	11.62	13.3
YOLOv5m	95.92	96.72	93.32	21.23	14.4
YOLOv5l	92.03	95.73	89.83	46.51	20.8
YOLOv5x	96.01	95.81	95.01	86.77	39.5

Analysis of the results in the table shows that the average accuracy of EET-YOLOV5 is 0.23% higher than YOLOv5m and 1.46% lower than YOLOv5x. The number of model parameters is 54.7%, 25%, and 13.4% of the YOLOv5m, YOLOv5l, YOLOv5x, respectively. And the detection speed is 1.1 ms, 7.5 ms, and 26.2 ms faster than the three, respectively.

Based on the results, it can be seen that the EET-YOLOv5 algorithm exceeds the detection accuracy of YOLOV5m, approaches the detection accuracy of YOLOv5x, and has an advantage in detection speed, despite having fewer parameters. Its application in clam meat detection scenarios can ensure the detection accuracy while achieving model lightweight.

Performance comparison of different target detection models

In order to prove the superiority of EET-YOLOv5 algorithm, it is compared with other algorithms. Table 3 shows the performance parameters of the four networks.

Table 3

Comparative experimental results

Model	Precision	Recall	mAP_0.5	Model parameter quantity	Detection speed
	[%]	[%]	[%]	[M]	[ms]
EET-YOLOv5	93.03	97.03	93.55	11.62	13.3
YOLOv4	86.47	91.20	91.13	61.05	14.1
Faster-RCNN	88.50	90.24	89.06	63.09	52.9
SSD	81.31	86.78	81.43	10.71	8.4

Analysis of the data in the table shows that the average accuracy of EET-YOLOv5 is improved by 2.42%,4.49%,12.12% compared to YOLOv4, Faster-RCNN and SSD, respectively. The number of model parameters is 19.0%, 18.4% and 108.4% of the three, respectively. The detection speed is improved by 0.8 ms, 39.6 ms and decreased by 4.9 ms over SSD compared to the former two respectively.

The analysis based on the results is as follows:

1) Compared with YOLOv4, YOLOv5 applies the CSP structure more widely in Backbone and Neck modules, and can calculate the anchor frame size automatically, which improves the detection speed and reduces the model parameters while guaranteeing the accuracy, and the improved EET-YOLOv5 has a greater improvement in accuracy.

2) Faster-RCNN as a two-stage detection algorithm, has a significantly lower detection speed than EET-YOLOv5, which is a single-stage detection algorithm. Since the detection algorithm proposed in this study is intended for use on a dynamically operating production line, real-time performance is critical. The low detection speed of Faster R-CNN would negatively impact production efficiency, making it unsuitable for this application scenario.

3) The lightweight detection algorithm SSD demonstrates advantages over EET-YOLOv5 in terms of fewer parameters and faster detection speed. However, its detection accuracy is significantly lower than that of EET-YOLOv5.

To sum up, EET-YOLOv5 is more suitable for clam meat detection. The comparison of average precision across different models is illustrated in Figure 6.

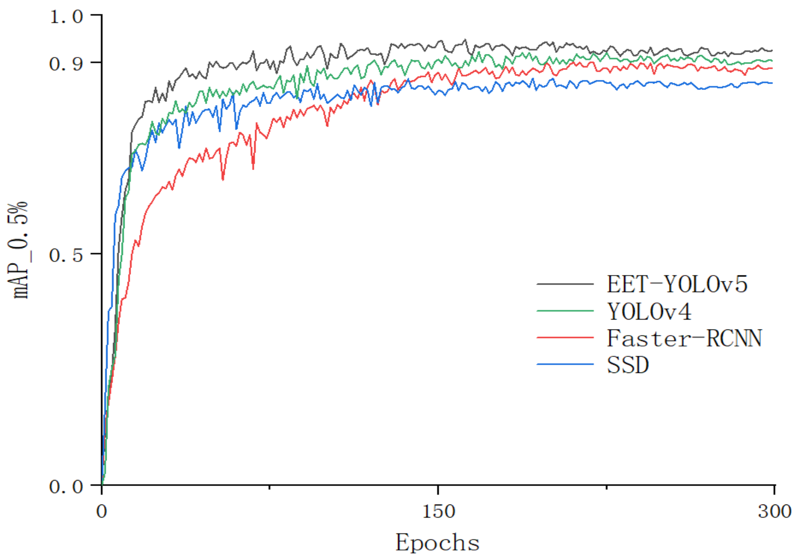


Fig. 6 - Mean average precision (mAP) curves of different target detection algorithms

Epochs is the number of training rounds

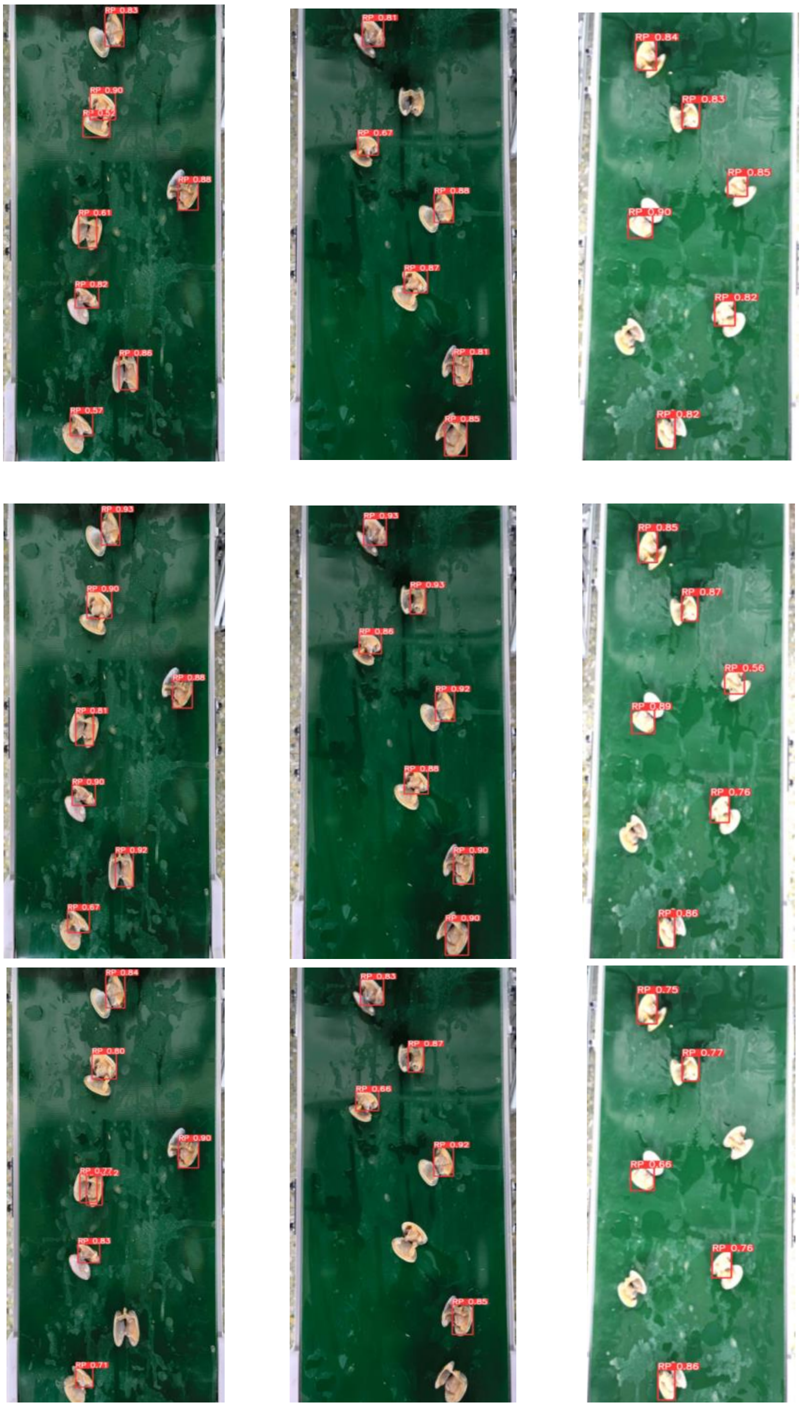
To provide a more intuitive comparison, the detection results of the four target detection algorithms on the same test set are visualized. The results are presented in Figure 7.

EET-YOLOv5

YOLOv4

Faster-RCNN

SSD



A B C

Fig. 7 - Results from various object detection networks

As shown in Figure 7, panels A and B represent detection results under normal illumination conditions. Under these conditions, YOLOv4 and SSD exhibit both missed and false detections, along with low confidence scores. While Faster R-CNN and EET-YOLOv5 successfully detect all targets, the confidence scores produced by Faster R-CNN are lower than those of EET-YOLOv5. Panel C shows detection performance under simulated strong indoor lighting. In this scenario, all detection networks except EET-YOLOv5 fail to detect one or more targets. EET-YOLOv5 successfully detects all clam meat instances with higher confidence than the other models. These comparative experiments demonstrate that EET-YOLOv5 achieves the best overall performance across varying lighting conditions. However, the experiments also revealed that in a few cases where clams were severely damaged during shelling, EET-YOLOv5 missed detections due to irregular shapes of the clam meat. This issue may be addressed by improving the shell-opening method and further optimizing the target detection network.

Workbench experiment

To validate the real-world deployment effectiveness of the proposed algorithm, a workbench experiment was conducted for clam meat positioning. A test bench was constructed, where a binocular vision system was used to capture the position parameters of the clam meat. Based on the detected coordinates, the system calculated the movement parameters for the XY synchronous belt, guiding a laser emitter to align precisely above the target. Successful laser irradiation of the clam meat indicated accurate positioning.

The experiment was conducted in the Marine Products Processing Laboratory at Shandong University of Technology. The setup is shown in Figure 8. A total of 120 positioning trials were carried out, achieving an average positioning time of 1.8 seconds and an average success rate of 92.7%, demonstrating the system's feasibility for practical application.

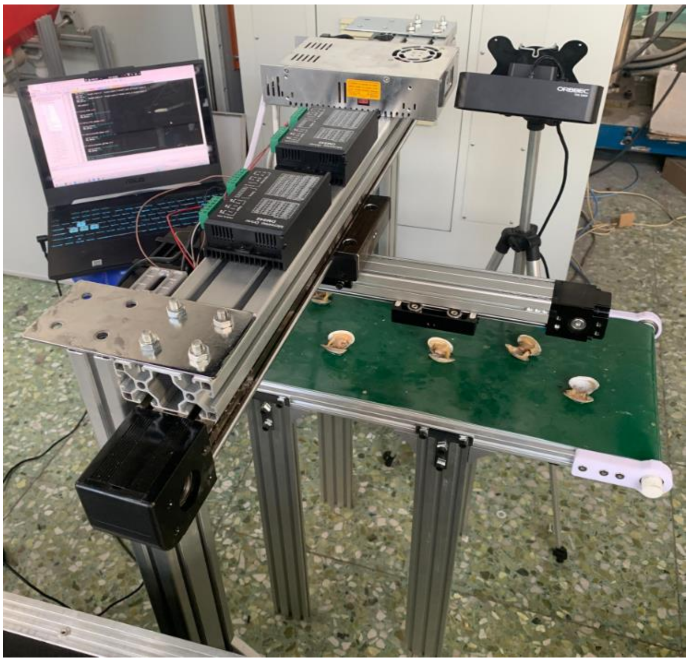


Fig. 8 - Workbench experiment

CONCLUSIONS

Intelligent shelling technology plays a key role in maintaining the freshness of clam meat while improving the efficiency of shell-meat separation, and it represents a promising development direction for the processing of *Ruditapes philippinarum*. In this study, an improved target detection network - EET-YOLOv5, based on YOLOv5 - was proposed for identifying clam meat on an automatic shelling production line.

Several enhancements were introduced to the original YOLOv5s architecture. The Efficient Local Attention (ELA) mechanism was added to the backbone network to suppress irrelevant information and enhance feature learning. The original CIoU loss function was replaced with the EIoU loss function to improve detection accuracy and network training speed. Additionally, the original detection head was replaced with the TSCODE decoupled head to improve both localization precision and classification accuracy. Ablation and comparative experiments were conducted to validate the effectiveness of these improvements. On a custom dataset, the proposed EET-YOLOv5 network achieved 93.03% precision, 97.03% recall, 93.55% mean average precision (mAP), and a detection speed of 13.3 ms, while maintaining high detection accuracy under varying lighting conditions. The main source of false detections was attributed to clam meat deformation caused by shelling, which can be addressed by improving the shell-opening process. The algorithm was deployed on a physical test bench for real-world positioning experiments, achieving an average positioning time of 1.8 seconds and a success rate of 92.7%. Although the model complexity of EET-YOLOv5 is slightly higher than that of the original YOLOv5s, the improvement in detection performance is substantial. Compared to other mainstream detection algorithms such as YOLOv4, Faster R-CNN, SSD, and models in the YOLOv5 series, EET-YOLOv5 shows clear advantages and is capable of meeting the demands of clam meat detection in production environments.

REFERENCES

[1] Chang Y.Q., (2008). *Shellfish enhancement* (水产养殖). China Agriculture Press., Chaoyang District, China.

[2] Dong Z. P., Yue X. X., Tian Z. X., Hou S.F., & Jiang L.S., (2023). Mussel detection algorithm based on YOLO (基于 YOLO 的贻贝检测算法). *Journal of Shanghai Ocean University. 32(5)*, 1015-1023. DOI: 10.12024/ sou.20230604238

[3] Feng Y. R., Zhang X. D., & Tao X. H., (2022). Recognition and Detection of Shellfish Based on Improved Faster R-CNN（基于改进的 Faster R-CNN 贝类识别检测）. *Journal of Dalian Polytechnic University. 41(3)*, 208-213. DOI: 10.19670/j.cnki.dlgydxxb.2022.0311

[4] Gevorgyan Z., (2022). SIoU loss: More powerful learning for bounding box regression. *arXiv:*2205.12740. DOI: 10.48550/arXiv.2205.12740

[5] He K., Zhang X., Ren S., & Sun J., (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence. 37(9)*, 1904-1916. DOI: 10.1109/TPAMI.2015.2389824

[6] Hou Q., Zhou D., & Feng J., (2021). Coordinate attention for efficient mobile network design. *IEEE/CVF conference on computer vision and pattern recognition.* pp. 13713-13722. DOI: 10.48550/arXiv.2103.02907

[7] Hu J., Shen L., Albanie S., Sun G., & Wu E., (2019). Squeeze-and-Excitation Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 42(8)*, 2011-2023. DOI: 10.1109/TPAMI.2019.2913372

[8] Liu S., Qi L., Qin H., Shi J., & Jia J., Path aggregation network for instance segmentation. *IEEE conference on computer vision and pattern recognition*. pp. 8759-8768. DOI: 10.48550/arXiv.1803.01534

[9] Lin T. Y., Dollár P., Girshick R., He K., Hariharan B., & Belongie S., (2017). Feature pyramid networks for object detection. *IEEE conference on computer vision and pattern recognition.* pp. 2117-2125. DOI: 10.48550/arXiv.1612.03144

[10] Liu Z L., Wang J., & Mai R G., (2025). Detecting underwater sea cucumber in marine ranching using improved YOLOv10s. *Transactions of the Chinese Society of Agricultural Engineering. 41 (10):* 186-194. DOI: 10.11975/j.issn.1002-6819.202412109.

[11] Rezatofighi H., Tsoi N., Gwak J., Sadeghian A., Reid I., Savarese S., (2019). Generalized intersection over union: A metric and a loss for bounding box regression. IEEE/CVF conference on computer vision and pattern recognition. pp. 658-666.

[12] Song G., Liu Y., & Wang X., (2020). Revisiting the sibling head in object detector. *IEEE/CVF conference on computer vision and pattern recognition.* pp. 11563-11572. DOI:10.48550/arXiv.2003.07540

[13] Wang H. F., & Zhang K., (2016). Research on Shellfish Image Classification Algorithm based on Gabor Features. *The Journal of New Industrialization*. 6(2):59-62.

[14] Wang Q., Wu B., Zhu P., Li P., Zuo W., Hu Q., (2020). ECA-Net: Efficient channel attention for deep convolutional neural networks. IEEE/CVF conference on computer vision and pattern recognition. pp. 11534-11542.

[15] Wu Y., Chen Y., Yuan L., Liu Z., Wang L., Li H., & Fu Y., (2020). Rethinking classification and localization for object detection. *IEEE/CVF conference on computer vision and pattern recognition.* pp. 10186-10195. DOI:10.48550/arXiv.1904.06493

[16] Wu Y., & He K., (2018). Group normalization. *European conference on computer vision (ECCV).* pp. 3-19. DOI:10.48550/arXiv.1803.08494

[17] Woo S., Park J., Lee J. Y., & Kweon I. S., (2018). Cbam: Convolutional block attention module. *European conference on computer vision (ECCV)*. pp. 3-19. DOI:10.48550/arXiv.1807.06521

[18] Xing C., An Y., Chen J., Kong J. Y., & Liu G. Z., (2021). Marine Stewardship Council's strategy and practice of enhanced fisheries: Taking Ruditapes philippinarum fishery as a case study. *Fishery Information and Strategy. 36(4)*, 245-251.

[19] Xu W., & Wan Y., (2024). ELA: Efficient local attention for deep convolutional neural networks. *arXi,* 2403.01123. DOI: 10.48550/arXiv.2403.01123

[20] Yu Z., Wen L T., & Sun J., (2025). Dense small target detection algorithm for freshwater snails based on improved YOLOV8-OBB. *Acta hydrobiologica sinica.* 1-12 (2025-07-16). DOI: 10.7541/2025.2024.0497

[21] Zhang J., Yi J. G., Jiang H. Y., Wang J. Z., & Liu J. T., (2013). Shelling technique optimization of scallop by steam (蒸汽处理扇贝的剥壳技术优化). *Guangdong Agricultural Sciences. 40(14)*, 120-122.

[22] Zhang Y., Yue J., Jia S. X., Li Z. B., & Sheng G.R., (2022). Recognition of Shellfish Based on Visible Spectrum and Convolutional Neural Network (基于卷积神经网络的贝类识别方法研究). *Spectroscopy and Spectral Analysis. 42(10)*, 3298-3306.

[23] Zhao Y., Zhu X., Zhao J. B., & Fang G. A., (2024). Research Progress on Shell-Meat Separation Technology of Shelled Aquatic Products: A Review (带壳类水产品的壳肉分离技术研究进展). *Journal of Dalian Ocean University. 38(6)*, 1083. DOI:10. 16535/ j. cnki. dlhyxb. 2023-059

[24] Zheng Z., Wang P., Liu W., Li J., Ye R., & Ren D., (2020). Distance-IoU loss: Faster and better learning for bounding box regression. *AAAI conference on artificial intelligence. 34(07)*, 12993-13000. DOI:10.1609/aaai.v34i07.69