doi:10.6041/j.issn.1000-1298.2026.02.026

Method for Behavior Recognition of Hu Sheep in Intensive Farming Based on HLNC - YOLO

JI Ronghua[1] CHANG Hongrui[1] ZHANG Suoxiang[1] LIU Zhongying[2] WU Zhonghong[2]
(1. *College of Information and Electrical Engineering*, *China Agricultural University*, *Beijing* 100083, *China*
2. *College of Animal Science and Technology*, *China Agricultural University*, *Beijing* 100093, *China*)

Abstract: Behavior recognition of Hu sheep contributes to their intensive and intelligent farming. Due to the generally high density of Hu sheep farming, severe occlusion occurs among different behaviors and even among sheep performing the same behavior, leading to missing and false detection issues in existing behavior recognition methods. A high-low frequency aggregated attention and negative sample comprehensive score loss and comprehensive score soft non-maximum suppression - YOLO (HLNC - YOLO) was proposed for identifying the behavior of Hu sheep, addressing the issues of missed and erroneous detections caused by occlusion between Hu sheep in intensive farming. Firstly, images of four typical behaviors—standing, lying, eating, and drinking—were collected from the sheep farm to construct the Hu sheep behavior dataset (HSBD). Next, to solve the occlusion issues, during the training phase, the C2F - HLAtt module was integrated, which combined high-low frequency aggregation attention, into the YOLO v8 Backbone to perceive occluded objects and introduce an auxiliary reversible branch to retain more effective features. Using comprehensive score regression loss (CSLoss) to reduce the scores of suboptimal boxes and enhance the comprehensive scores of occluded object boxes. Finally, the soft comprehensive score non-maximal suppression (Soft - CS - NMS) algorithm filtered prediction boxes during the inferencing. Testing on the HSBD, HLNC - YOLO achieved a mean average precision (mAP@50) of 87.8%, with a memory footprint of 17.4 MB. This represented an improvement of 7.1, 2.2, 4.6, and 11 percentage points over YOLO v8, YOLO v9, YOLO v10, and Faster R - CNN, respectively. Research indicated that the HLNC - YOLO accurately identified the behavior of Hu sheep in intensive farming and possessed generalization capabilities, providing technical support for smart farming.

Key words: behavior recognition; YOLO; loss function; attention mechanism

CLC number: TP391.41 **Document code**: A **Article ID**: 1000-1298(2026)02-0265-11 **OSID**:

0 Introduction

As the main livestock animal in China, sheep hold a vital position in the livestock industry production, characterized by large production scale and wide geographical distribution, playing significant roles in food supply and promoting economic development. The Hu sheep is a high-quality meat sheep breed in China. It is known for its reproductive solid capacity, preference for quiet, shady, and dry environments, and suitability for intensive and large-scale farming[1-3]. Hu sheep are kept in relatively dark sheep pens throughout their lives, and the behavioral changes they exhibit during rearing can reflect their overall health status[4]. As the scale of farming expands, there is an increasing demand for the accuracy and real-time monitoring of Hu sheep behavior. Traditional manual monitoring methods are no longer sufficient, and an urgent need is to replace them with automated detection methods.

Research on poultry and livestock behavior recognition methods was divided into contact-based and non-contact-based approaches. Contact-based methods involved attaching contact sensors to animals to obtain

Received date: 2024-09-13 Accepted date: 2024-10-22
Foundation item: National Key Research and Development Program of China (2022YFD1301104)
First author: JI Ronghua, associate professor. E-mail: jessic1212@cau.edu.cn
Corresponding author: WU Zhonghong, professor. E-mail: wuzhh@cau.edu.cn

multidimensional information, such as triaxial acceleration, angular velocity, temperature, and sound, and then using machine learning methods to model this data for observing animal behaviors[5-6]. However, contact-based methods can affect animals' normal growth, cause stress reactions, and have a short sensor lifespan with high costs.

Non-contact behavioral recognition methods collected video images, using techniques like deep learning to process images, extract features, and ultimately achieve the recognition of animal behavior in livestock farming. They were widely used in intensive animal farming because they did not affect the animals' daily lives. Typically, poultry and livestock behavior recognition were divided into single animal or single behavior recognition and multiple animal multiple, behavior recognition.

Regarding single animal or single behavior recognition, CHENG et al.[8] collected behavioral data from sheep and accurately recognized standing, lying, feeding, and drinking behaviors by using YOLO v5 based on the overhead angle. XUE et al.[9] improved the YOLO v5s to achieve heat detection of sows. MA et al.[10] proposed a cow behavior recognition method based on ReXNet 3D, achieving recognition of individual cows' lying, standing, and walking behaviors.

In terms of multiple animals and multiple behavior recognition. HU et al.[11] added CBAMs attention modules on YOLO v5 to achieve the recognition of standing, walking, and lying of sheep in open grasslands, with 3 ~ 5 sheep in a single frame. WANG et al.[12] achieved recognition of five behaviors of standing, lying, walking, feeding, and drinking of sheep in pens and activity squares based on an improved YOLO v8s model, with only slight occlusion in the pen data. DUAN et al.[13] used SNSS - YOLO v7 to recognize seven behaviors of beef cattle in outdoor monitoring, with approximately 3 ~ 10 cattle in a single image, with slight occlusion. SONG et al.[14] achieved recognition of occluded beef cattle targets through ECA - YOLO v5s, but the occlusion issues in the dataset were singular. BAI et al.[15] improved the YOLO v5s network by simultaneously detecting the heads of cows and water troughs and feed troughs to achieve recognition of six basic behaviors of cows, including walking, standing, lying , crawling, drinking, and eating, with approximately 3 ~ 10 cows in the image, but no occlusion issues as the behavior data were collected in outdoor pastures. RONG et al.[16] implemented the detection of crawling behavior in a herd of cattle based on a lightweight improvement of YOLO v5, but the data showed that the cattle herd was not densely distributed.

In summary, current research on identifying poultry and livestock behavior focused on multi-target and multi-behavior recognition. Research scenarios were primarily in outdoor open environments, with only slight obstructions between animals. More research was needed on animal behavior recognition in densely farmed closed environments.

A typical Hu sheep behavior recognition method was proposed based on HLNC - YOLO to address the severe occlusion issues between Hu sheep in intensive farming, leading to missed and false detections. HLNC - YOLO recognized four behaviors: standing, lying, eating, and drinking. HLNC - YOLO introduced the HLAttention module to extract fine-grained features and enhance the scores of occluded targets; it used the CSLoss function to reduce the scores of suboptimal boxes; then in the model inference stage, the Soft - CS - NMS algorithm was used to filter prediction boxes to retain occluded targets; finally, an auxiliary reversible branch was introduced to ensure HLNC - YOLO retained more effective features during training. Providing technological support for intelligent Hu sheep farming.

1 Materials and methods

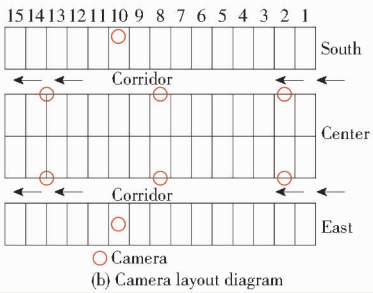
1.1 Data collection

This experiment used the Anxin sheep farm in Bozhou, Anhui, as the experimental site to collect video data. The farm was divided into 60 semi-open sheep pens, covering an area of 1 440 m^2, with a stocking density of 0.63/m^2 and nipple-style drinkers installed on the wall of each pen. The everyday behaviors of the sheep, such as standing, lying, eating, and drinking, were all carried out within the pens. Video cameras (Dahua 1230C - A, focal length 2.8 mm, resolution 1080p, video frame rate 50 Hz) were installed on the roof supports of the pens. The cameras used IR - CUT to switch between visible light

and infrared sensitivity modes, covering the two pens in front of them at an oblique downward angle. The farm site was depicted in Fig. 1a, with its floor plan and camera layout shown in Fig. 1b, where each square represented a sheep pen and red circles mark the camera placement points. Each camera operated around the clock, saving video data every 0.5 h, and had collected raw video footage from August 11, 2023, to March 11, 2024, spanning 212 days, with 5 088 video segments.



(a) Farm site



(b) Camera layout diagram

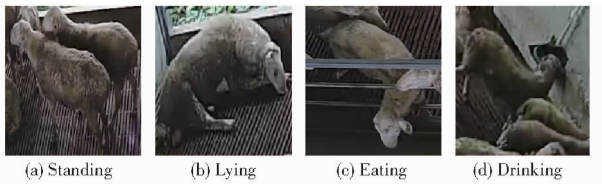
Fig. 1　Farm layout and camera distribution

1.2　Hu sheep behavior dataset

Totally 30 daytime videos and 25 nighttime videos for the four behaviors of Hu sheep were manually selected: standing, lying, eating, and drinking. The selected videos came from different dates. Images were extracted at one-minute intervals for static and thirty-second intervals for dynamic actions. Images with high similarity were filtered out. A total of 400 daytime images and 300 nighttime images were obtained and divided into training and validation sets in an 8∶2 ratio. Data augmentation was applied to the images using local blurring and varying light intensity. The final dataset consisted of a training set composed of 960 daytime images and 720 nighttime images and a validation set composed of 240 daytime images and 180 nighttime images, forming the behavior recognition dataset for this study. The typical behavior determinations of Hu sheep and the behavior quantities in the training set were shown in Table 1, with behavior examples shown in Fig. 2.

Table 1　Description Hu sheep behavior dataset

Behavior	Behavior description	Behavior count
Standing	Four limbs support body standing, and the head does not touch the ground.	9 139
Lying	The limbs bend and touch the ground.	13 284
Eating	Eating with head bowed at the feeding trough on all fours	3 748
Drinking	Drinking with head lowered by the water pipe on all fours	51



(a) Standing　(b) Lying　(c) Eating　(d) Drinking

Fig. 2　Diagrams of different behaviors of Hu sheep

1.3　Related work

After YOLO v1, YOLO v2, and YOLO v3 determined the detection architecture of the Backbone, Neck, and Head, the YOLO algorithms[17－21] gradually became mainstream for object detection. YOLO v4[22] and YOLO v5[23] introduced the CSPNet[20] design to replace DarkNet[21]. YOLO v6[24] proposed BiC and SimCSPPPF. YOLO v7[25] introduced E－ELAN with rich gradient. YOLO v8[26] proposed the C2f module for feature extraction and fusion. Gold－YOLO[27] provided the GD mechanism to enhance the multi-scale feature fusion ability. YOLO v9[28] proposed the GELAN improved architecture with PGI to enhance the training process. YOLO v10[29] proposed a training method for one-to-many detection heads, assisting one-to-one detection heads. The decoupled head was widely used in object detection[30－31]. During the YOLO model inference stage, the YOLO model generated multiple prediction boxes around each pixel point and then used NMS for filtering. In the actual breeding environment of Hu sheep, the breeding density was relatively high, and occlusion issues were common, as shown in Fig. 3. The NMS algorithm was prone to misdetections during filtering.



(a) Slight occlusion　(b) Moderate occlusion　(c) Severe occlusion

Fig. 3　Different occlusion conditions for Hu sheep

Based on the model inference process, occlusion

issues could be categorized into two types. The first type involved occlusions of Hu sheep by other Hu sheep and by the environment. During model training, occluded targets were assigned positive samples, but due to significant information loss, their classification scores were low, leading to their exclusion during score threshold filtering. The second type of occlusion issues involved occlusions among Hu sheep exhibiting the same behavior. When performing IOU threshold filtering within the behavior category, occluded targets were treated as "sub-optimal boxes" of non-occluded targets and thus filtered out.

2 Hu sheep behavior recognition model

2.1 HLNC – YOLO architecture

HLNC – YOLO model architecture was shown in Fig. 4. The original image was resized to 640 pixels × 640 pixels as the model input. The Backbone extracted three scaled feature maps of 80 pixels × 80 pixels, 40 pixels × 40 pixels, and 20 pixels × 20 pixels; corresponding three scaled features were obtained through feature fusion at the Neck; an auxiliary reversible branch was introduced; detection results were obtained by using the YOLO v8 detection head. DFLoss, CIoULoss, and BCELoss were applied to positive samples, CSLoss was applied to negative samples, and the optimizer adjusted the model parameters. During the inference stage of the model, the CS – Soft – NMS algorithm was used to filter the predicted boxes.

2.2 HLNC – YOLO structure

The first type of occlusion issues was shown in Fig. 5, because the feature extraction method of convolution cannot focus on fine-grained features, causing a low score for target 4 to be missed. Therefore, an HLAttention attention method was proposed to focus on pixel-level information, a C2F – HLAtt was constructed to fuse feature map with different frequency information, and finally the HLNC – YOLO model network structure was established to achieve information extraction of occluded targets and improve the classification scores of occluded targets.

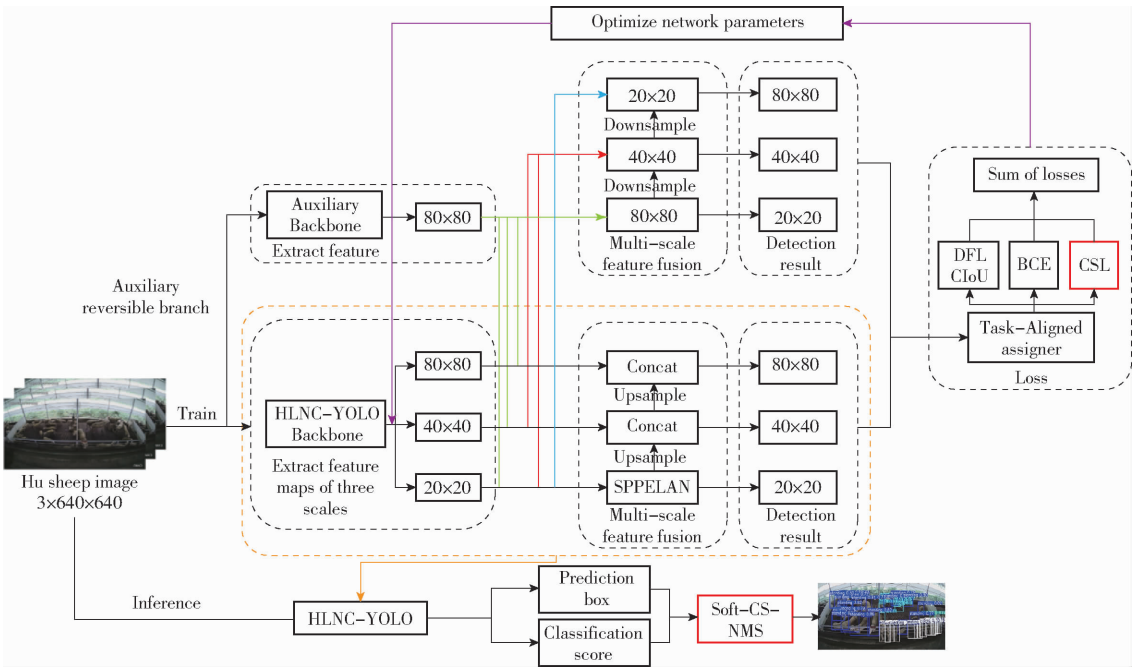


Fig. 4 Hu sheep behavior recognition model architecture

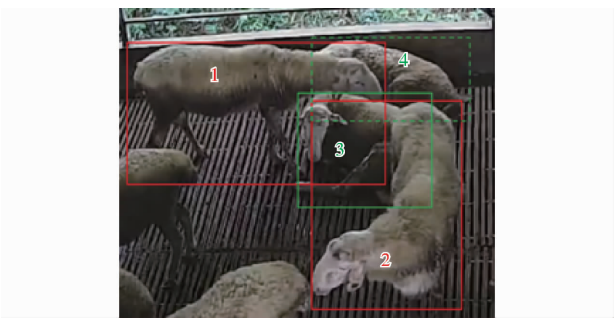


Fig. 5 First type of occlusion issues

2.2.1 High-low frequency aggregated attention (HLAtt)

Inspired by the functions of biological visual[32], a pixel-focused attention mechanism called High-low frequency aggregated attention was proposed. Treating pixels as query achieved fine-grained focused

perception while maintaining perception at a coarse granularity. This allowed for attention to be placed on the high and low frequency information of pixels in feature maps, as shown in Fig. 6.

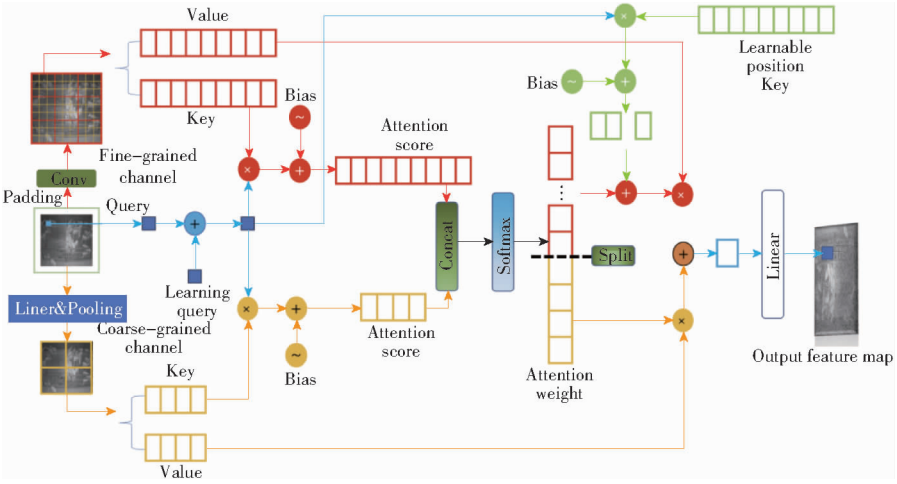


Fig. 6　High-low frequency aggregated attention

For a given image, HLAtt took the pixel (i, j) as query ($Q_{(i,j)}$), and a sliding window area of size $k \times k$, $|\rho(i,j)| = k^2$, as the source of key ($\boldsymbol{K}_{\rho(i,j)}$) and value ($V_{\rho(i,j)}$) for fine-grained attention. And HLAtt extracted features from a fixed window area of size $b \times b$, $|\sigma(X)| = b^2$, as the source of key ($\boldsymbol{K}_{\sigma(X)}$) and value ($V_{\sigma(X)}$) for coarse-grained attention. To enhance spatial position awareness, two learnable positional encodings were introduced: learnable query (LQ) and learnable positional key (LPK). Finally, length-scaled cosine attention was used to generate more suitable attention weights.

Regarding position bias, HLAtt directly learned the position offset $B(i, j)$ on the sliding window path and used the log-spaced continuous position bias (log－CPB) to include spatial relative coordinates to represent coarse-grained information on the fixed window path.

In the computation of the entire HLAtt, the attention scores for fine-grained ($S_{\rho(i,j)}$) and coarse-grained ($S_{\sigma(X)}$) were calculated as formula (1).

$$\begin{cases} S_{\rho(i,j)} = (Q_{(i,j)} + LQ)\boldsymbol{K}^{\mathrm{T}}_{\rho(i,j)} \\ S_{\sigma(X)} = (Q_{(i,j)} + LQ)\boldsymbol{K}^{\mathrm{T}}_{\sigma(X)} \end{cases} \tag{1}$$

The positional biases were directly concatenated in a one-to-one correspondence with the queries of both fine-grained and coarse-grained, as shown in formula (2).

$$B_{(i,j)} = Concat(B_{(i,j)\sim\rho(i,j)}, \lg - CPB(\Delta_{(i,j)\sim\sigma(X)})) \tag{2}$$

where $\Delta_{(i,j)\sim\sigma(X)}$ was the spatial relative coordinates between $Q_{(i,j)}$ and $\boldsymbol{K}_{\sigma(X)}$.

Coupling $S_{\rho(i,j)}$ and $S_{\sigma(X)}$ and calculating the attention weights ($A_{(i,j)}$) by using the softmax function. By splitting the coupled attention weights, fine-grained weights ($A_{\rho(i,j)}$) and coarse-grained weights ($A_{\sigma(X)}$) can be obtained, as shown in formula (3) and formula (4).

$$A_{(i,j)} = softmax(\tau \lg NConcat(S_{\rho(i,j)} + S_{\sigma(X)}) + B_{(i,j)}) \tag{3}$$

$$A_{\rho(i,j)}, A_{\sigma(X)} = Split(A_{(i,j)}) \tag{4}$$

Equation (4) yielded the attention result. The "," was used to separate the two parts resulting from the split operation. $A_{\rho(i,j)}$ and $A_{\sigma(X)}$ denote the two partitioned components of the attention output resulting from the decomposition of $A_{(i,j)}$.

Multiplying the attention scores of $A_{\rho(i,j)}$ and $A_{\sigma(X)}$ with their corresponding values ($V_{\rho(i,j)}$ and $V_{\sigma(X)}$) and summing them up. Then obtained the result at position (i, j) in the new feature map through a fully connected layer. As shown in formula (5) and formula (6).

$$HLA(i,j) = (A_{\rho(i,j)} + Q_{(i,j)}LPK)V_{\rho(i,j)} + A_{\sigma(X)}V_{\sigma(X)} \tag{5}$$

$$F(i,j) = f(HLA(i,j)) \tag{6}$$

where $HLA(i,j)$ was the result of HLAtt, and $f(X)$ was a fully connected layer computation.

2.2.2　C2F－HLAtt

The C2F module proposed by YOLO v8 divided the input feature map into two parts at each step, with one part undergoing no transformation and the other undergoing convolution operation to extract features and finally merging all features. Using HLAtt to replace the convolution, proposing the C2F－HLAtt module, which integrated the extraction and fusion of high-low

frequency features, and the whole module achieved the fusion of high and low-dimensional features. The structure of the C2F – HLAtt module was shown in Fig. 7.

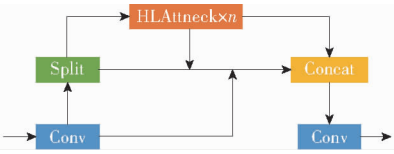


Fig. 7 C2F – HLAtt structure

2.2.3 HLNC – YOLO overall structure

The HLNC – YOLO was shown in Fig. 8. In Backbone, HLAtt was used to extract original image information, and then the C2F module responsible for extracting different scale features in YOLO v8 was replaced with the C2F – HLAtt module to focus on fine-grained features. This allowed the model to simultaneously consider feature extraction and fusion of different frequencies, dimensions, and scales when facing occlusion issues in order to enhance the model's ability to extract information from obscured targets.

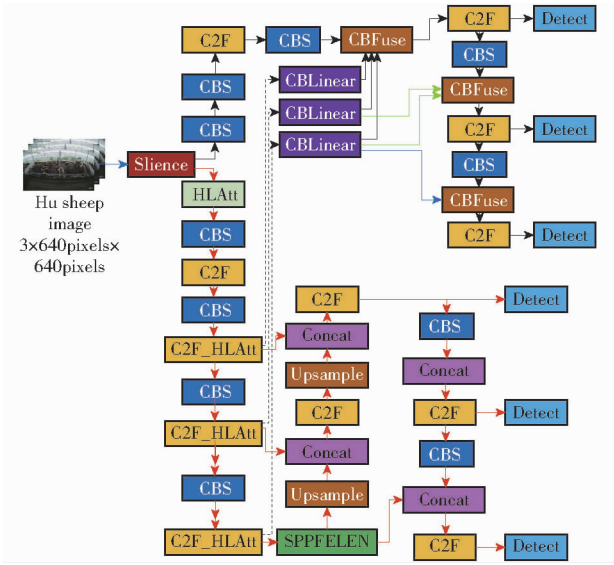


Fig. 8 HLNC – YOLO network structure

2.3 Loss function and post-processing method

The second type of occlusion issues was shown in Fig. 9. After model inference, the NMS algorithm, at the standard of mAP@50, treated the dashed box as a suboptimal box and missed the detection. Therefore, a comprehensive score loss (CSLoss) was addedto the negative samples to reduce the negative boxes' score and increase occluded targets' competitiveness. On the other hand, the Soft – CS – NMS algorithm was used for box filtering during prediction to retain prediction boxes greater than the threshold.

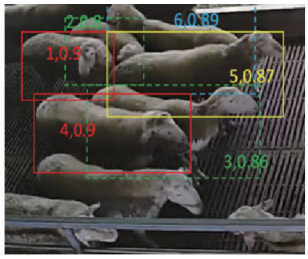


Fig. 9 Second type of occlusion issues

2.3.1 Comprehensive score regression loss(CSLoss)

The output of the Head was divided into two parts. One part was the classification score of the predicted box, and the other part described the expected value of the probability distribution of the distance from the four edges of the predicted box to the center point to define the boundary box.

Firstly, the size of the boundary box was evaluated by summing up the expectations, denoted as S. As shown in formula (7).

$$S = \sum_{i=0}^{3} \left(\sum_{j=0}^{reg_max} y_j P(y_j) \right) \tag{7}$$

where y was the distance, with values ranging from $(0, reg_max)$, i was the four extension directions (up, down, left, right) from the center point, $P(y_j)$ was probability of the bounding box reaching a distance y_j from the center point in the i-th direction, and reg_max was maximum distance from each side of the bounding box to the center point. The default value of reg_max was 15.

Then, comprehensive score (CS) was calculated based on classification score and S, as shown in formula (8).

$$CS = (\max(score))^{\alpha} \left(\frac{S}{4reg_max} \right)^{\beta} \tag{8}$$

where $score$ was probability of predicted box being classified into each of the four behaviors, α and β were the weights.

Finally, CSLoss was calculated based CS, as shown in formula (9).

$$CSLoss = \frac{S}{4reg_max}(0 - CS)^2 \tag{9}$$

2.3.2 Soft comprehensive score NMS (Soft – CS – NMS)

If the prediction boxes used the Soft – NMS algorithm[33], multiple detections were highly likely to occur in scenarios with severe occlusion. Therefore, the Soft – CS – NMS was introduced to preserve the occluded prediction boxes. By combining it with CSLoss, the model can widen the comprehensive score

difference between occluded targets and suboptimal boxes during the inference phase, allowing for a higher threshold to distinguish between the two. Fig. 10 showed the pseudocode for the Soft－CS－NMS algorithm.

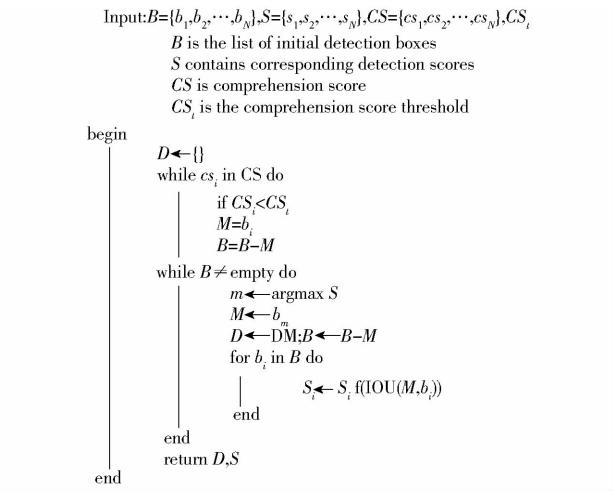


Fig. 10　Pseudocode for Soft－CS－NMS algorithm

2.4　Training parameter settings and evaluation

During training, the network was trained individually without using pre-trained weights and was run on the same server. Its configuration was as follows: the operating system was Ubuntu 20.04; GPU was RTX 3090 (24 GB); CPU was Intel(R) Xeon(R) Platinum 8255C with a central frequency of 2.50 GHz; memory was 43 GB, deep learning library was PyTorch 2.0.0, computing platform was Cuda 11.8, and the programming language was Python 3.8.

The hyperparameters for model training were as follows: the training optimizer was SGD, the epoch was 300, the batch size was 16, the initial learning rate was 0.001, and the weight decay was set to 0.000 5.

The performance of different models was assessed by using precision, recall, AP, and mAP@50. Among these, AP represented the average precision for recognizing various behaviors of Hu sheep, and mAP@50 was the average value of AP at the IOU threshold of 0.5.

This study evaluated model efficiency using parameters, FLOPs, amount of storage required, FPS, high-contribution area ratio R, and effective receptive field (ERF). FLOPs denoted the number of floating-point operations per second; FPS indicated the number of frames the model can process per second; R was defined as a minimum rectangle that covered the contribution scores over a given threshold t; the effective receptive field was the superposition of practical values in batches and channels.

3　Results

3.1　Different behavior recognition

Considering frequency, dimensionality, and scale during feature extraction and fusion, enhancing HLNC－YOLO ability to perceive targets with missing information such as occlusions and blurriness. To verify the recognition effectiveness of HLNC－YOLO for identifying four typical behaviors of Hu sheep, it was compared with the YOLO v8, and the results were shown in Table 2.

Table 2　Comparison result of mAP@50 of different targets

Behavior	Behavior count	mAP@50/%	
		HLNC－YOLO	YOLO v8
Standing	1 675	95.5	93.7
Lying	2 924	95.8	94.6
Eating	759	95.1	95.0
Drinking	15	64.6	38.5
All	5 373	87.8	80.7

Table 2 showed that HLNC－YOLO improved the recognition of four typical behaviors. The mAP@50 for these four typical behaviors was increased by 1.8, 1.2, 0.1, and 26.1 percentage points, respectively, with the most considerable improvement in the drinking behavior. Overall, the mAP@50 was enhanced by 7.1 percentage points. Drinking behaviors occurred less frequently and were located at the image edges, with drinking troughs and drinking sheep being relatively blurry. This result validated the capability of HLNC－YOLO to extract fine-grained information.

To further verify that HLNC－YOLO simultaneously considered pixel-wise contextual information and global information, the high-contribution area ratio for the output feature maps of the last layer C2F－HLAtt of Backbone and the output feature maps at the same position C2f in YOLO v8 were calculated, as shown in Table 3. Fig. 11 visualized both models' ERF.

As shown in Table 3, at each contribution threshold, the high-contribution area ratio of HLNC－YOLO was greater than that of YOLO v8, indicating that HLNC－YOLO can perceive a larger area of information from the original image without using large

Table 3 High-contribution area ratio of different model

Model	Threshold	Area	R
HLNC – YOLO	0.2	137	0.075 82
YOLO v8		129	0.045 82
HLNC – YOLO	0.3	173	0.073 06
YOLO v8		165	0.066 46
HLNC – YOLO	0.5	251	0.153 81
YOLO v8		237	0.137 13
HLNC – YOLO	0.99	621	0.941 51
YOLO v8		541	0.714 55

kernel convolutions. Fig. 11 showed that while focusing on the center, HLNC – YOLO had a more extensive divergence than YOLO v8, further verifying that HLAtt maintained the ability to perceive over a more extensive range while focusing on fine-grained features. This also confirmed the ability of HLNC – YOLO to extract detailed features in dense and blurred conditions.

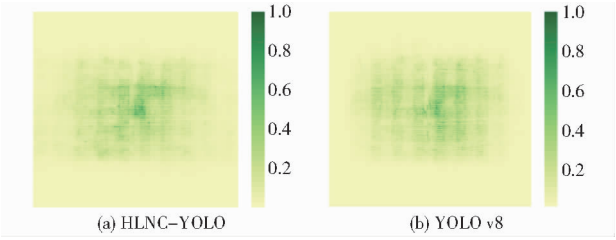


Fig. 11 HLNC – YOLO and YOLO v8 ERF

3.2 Ablation experiment

To further investigate the effectiveness of HLNC – YOLO improvements on YOLO v8, ablation experiments were designed to examine various aspects, including the Backbone, Attention, Loss, Soft – CS – NMS algorithm, and auxiliary reversible branch. After training each network, the auxiliary reversible branch was removed. The results of the ablation experiment on the test set were shown in Table 4.

As shown in Table 4, using C2F – HLAtt in Backbone improved mAP @ 50 by 2.8 percentage points; using CSLoss improved mAP @ 50 by 2 percentage points; employing the Soft – CS – NMS algorithm during inference enhanced mAP@ 50 by 1.2 percentage points. Using auxiliary reversible branch during model training improved mAP @ 50 by 2.3 percentage points. Test results showed that HLNC – YOLO was better at perceiving occluded and blurred targets, and the Soft – CS – NMS algorithm also detected a few missed targets during.

Table 4 Results of ablation experiments

Model	mAP@ 50/%	Parameters	FLOPs	Model size/MB
YOLO v8	80.7	3.01×10^6	8.1×10^9	14.4
YOLO v8 + C2F – HLAtt	84.5	3.15×10^6	8.5×10^9	15.3
YOLO v8 + C2F – HLAtt + CSLoss	86.5	3.15×10^6	8.5×10^9	15.3
HLNC – YOLO(without Soft – CS – NMS)	86.6	3.30×10^6	9.2×10^9	17.4
HLNC – YOLO (without auxiliary)	85.5	3.30×10^6	9.2×10^9	17.4
HLNC – YOLO	87.8	3.30×10^6	9.2×10^9	17.4

3.3 Comparison experiment

To verify the effectiveness of HLNC – YOLO model in recognizing the behavior of Hu sheep, it was compared with Faster R – CNN, YOLO v8, YOLO v9, and YOLO v10, using the same training and validation set. Table 5 showed the recognition effects of different models on goat behavior.

Table 5 showed that Faster R – CNN had the highest memory usage, parameters, and FPS but the worst recognition performance. Among YOLO models,

Table 5 Comparison of performance between different models

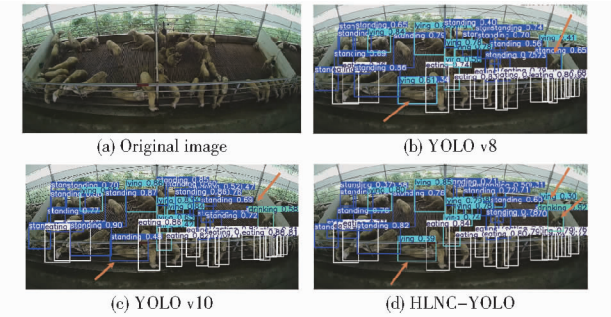
Model	mAP@ 50/%	Parameters	Model size/MB	FPS/(f·s^{-1})
Faster R – CNN	76.8	1.36×10^8	521.9	46.9
YOLO v5	78.9	2.08×10^7	71.4	65.4
YOLO v7	80.1	3.72×10^7	71.3	80.1
YOLO v8	80.7	3.01×10^6	14.4	101.1
YOLO v9	85.6	7.32×10^6	32.5	77.5
YOLO v10	83.2	2.27×10^6	10.5	146.3
HLNC – YOLO	87.8	3.30×10^6	17.4	69.2

YOLO v10 achieved the highest FPS by using One-to-One Heads for predictions, avoiding post-processing. HLNC – YOLO used Soft – CS – NMS, detecting fewer occluded targets but increasing post-processing time. With a slight increase in model parameters, HLNC – YOLO significantly improved mAP @ 50. Therefore, HLNC – YOLO had better overall performance.

4 Discussion

To verify the effectiveness of HLNC – YOLO in

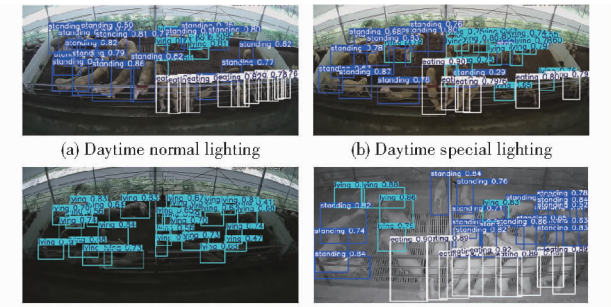
detecting obscured targets, the recognition effect of scenes with a high crowding density was tested and the results were compared with the YOLO v8 model. Fig. 12 showed that in heavily obscured areas, HLNC－YOLO assigned higher classification scores to obscured targets, detecting missed targets of YOLO v8. YOLO v8 mistakenly identified a sheep as both standing and laying down, and incorrectly identified drinking behavior as standing. The YOLO v10 correctly recognized the drinking behavior, but incorrectly identified lying down as standing, and missed the lying down target on the upper right.



(a) Original image　(b) YOLO v8　(c) YOLO v10　(d) HLNC－YOLO

Fig. 12　Behavior recognition results of Hu sheep

In addition, to verify the HLNC－YOLO's practicality, tests were conducted on the recognition of different lighting levels between day and night, as shown in Fig. 13. The model could adapt to various lighting environments and accurately recognize behavior in dim and nighttime conditions.



(a) Daytime normal lighting　(b) Daytime special lighting　(c) Evening　(d) Night

Fig. 13　Behavior recognition results of Hu sheep during day and night

5　Conclusion

Focusing on analyzing two types of occlusion issues for Hu sheep on the farm. Targeted solutions were proposed from the perspectives of network and post-processing, achieving good results.

(1) To solve the issue of the first type of occlusion of sheep in intensive farming, HLAtt was proposed to extract high and low-frequency information in the feature map, the C2F－HLAtt module was constructed to simultaneously extract and fuse different dimensional features, features of different scales was fused, and finally the auxiliary reversible branch was introduced to assist model feature extraction in HLNC－YOLO, enhancing the model's ability to extract information from obscured targets. The model parameters were only increased by 4.6%, resulting in mAP@50 increase of 5.8 percentage points.

(2) To solve the second occlusion issue encountered in object detection, CSLoss was proposed to calculate the loss for negative samples and the Soft－CS－NMS post-processing method was applied in the inference stage to retain occluded targets. The mAP@50 was increased by 1.2 percentage points without increasing the model parameters or memory consumption. However, this method increased the post-processing time for predicted bounding boxes in a single frame, decreasing FPS.

The results showed that HLNC－YOLO successfully addressed the occlusion of animals in intensive farming. It also considered the lighting conditions in farming scenarios, achieving good recognition results.

References

[1] ZHOU X, JIANG P. Production plan for captive Hu sheep based on planning model[C]//2024 IEEE 6th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). IEEE, 2024: 1966－1970.

[2] YUE G H. Reproductive characteristics of Chinese Hu sheep[J]. Animal Reproduction Science, 1996, 44(4): 223－230.

[3] LIU Z, ZHANG N, WEN Z, et al. Genomic insights into the origin, high fecundity and environmental adaptation of hu sheep [J]. Advanced Science, 2025, 12(37): e06492.

[4] WANG Kaiying, ZHAO Xiaoyang, HE Yong. Review on noninvasive monitoring technology of poultry behavior and physiological information[J]. Transactions of the CSAE, 2017, 33(20): 197－209. (in Chinese)

[5] CHEN G P, LI C, GUO Y, et al. Recognition of cattle's feeding behaviors using noseband pressure sensor with machine learning [J]. Frontiers in Veterinary Science, 2022, 9: 822621.

[6] ZHANG Xiyu, WU Pei, XUAN Chuanzhong, et al. Recognition of the movement behavior of study rams based on acceleration sensor [J]. Journal of China Agricultural University, 2018,23(11): 104－114. (in Chinese)

[7] WANG Shaohua, HE Dongjian. Estrus behavior recognition of dairy cows based on improved YOLO v3 model [J]. Transactions of the Chinese Society for Agricultural Machinery, 2021,52(7): 141 – 150. (in Chinese)
[8] CHENG M, YUAN H, WANG Q, et al. Application of deep learning in sheep behaviors recognition and influence analysis of training data characteristics on the recognition effect[J]. Computers and Electronics in Agriculture, 2022,198: 107010.
[9] XUE Hongxiang, SHEN Mingxia, LIU Longshen, et al. Estrus detection method of parturient sows based on improved YOLO v5s [J]. Transactions of the Chinese Society for Agricultural Machinery, 2023, 54(1): 263 – 270. (in Chinese)
[10] MA S, ZHANG Q, LI T, et al. Basic motion behavior recognition of single dairy cow based on improved Rexnet 3D network [J]. Computers and Electronics in Agriculture, 2022, 194: 106772.
[11] HU T, YAN R, JIANG C, et al. Grazing sheep behaviour recognition based on improved YOLO v5[J]. Sensors, 2023,23(10): 4752.
[12] WANG Wang, WANG Fushun, ZHANG Weijin, et al. Sheep behavior recognition method based on improved YOLO v8s [J]. Transactions of the Chinese Society for Agricultural Machinery, 2024,55(7):325 – 335,344. (in Chinese)
[13] DUAN Qingling, ZHAO Zhiqing, JIANG Tao, et al. Behavior recognition method of beef cattle based on SNSS – YOLO v7 [J]. Transactions of the Chinese Society for Agricultural Machinery, 2023,54(10):266 – 274,347. (in Chinese)
[14] SONG Huaibo, LI Rong, WANG Yunfei, et al. Recognition method of heavily occluded beef cattle targets based on ECA – YOLO v5s [J]. Transactions of the Chinese Society for Agricultural Machinery, 2023, 54(3): 274 – 281. (in Chinese)
[15] BAI Qiang, GAO Ronghua, ZHAO Chunjiang, et al. Multi-scale behavior recognition method for dairy cows based on improved YOLO v5s network[J]. Transactions of the CSAE, 2022, 38(12): 163 – 172. (in Chinese)
[16] WANG R, GAO R, LI Q, et al. A lightweight cow mounting behavior recognition system based on improved YOLO v5s[J]. Scientific Reports,2023,13(1):17418.
[17] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
[18] REDMON J, FARHADI A. YOLO9000: better, faster, stronger[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
[19] REDMON J, FARHADI A. YOLO v3: an incremental improvement[J]. arXiv Preprint, arXiv:1804.02767, 2018.
[20] WANG C Y, MARK LIAO H Y, WU Y H, et al. CSPNet: a new backbone that can enhance learning capability of CNN[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020.
[21] WANG C Y, BOCHKOVSKIY A, LIAO H Y M. Scaled-yolov4: scaling cross stage partial network[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: 13029 – 13038.
[22] BOCHKOVSKIY A, WANG C Y, LIAO H Y. YOLO v4: optimal speed and accuracy of object detection[J]. arXiv Preprint, arXiv:2004.10934, 2020.
[23] JOCHER G, CHAURASIA A, STOKEN A, et al. Ultralytics/yolov5: v7.0-yolov5 sota realtime instance segmentation[J/OL]. Zenodo, 2022. https://github.com/ultralytics/yolov5.
[24] LI C, LI L, GENG Y, et al. Yolov6 v3.0: a full-scale reloading[J]. arXiv Preprint, arXiv:2301.05586, 2023.
[25] WANG C Y, BOCHKOVSKIY A, LIAO H Y. Yolov7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023:7464 – 7475.
[26] JOCHER G, CHAURASIA A, QIU J. Ultralytics YOLO v8[CP/OL]. Version 8.0.0. ,2023. https://github.com/ultralytics/ultralytics.
[27] WANG C, HE W, NIE Y, et al. Gold-yolo: efficient object detector via gather-and-distribute mechanism[J]. Advances in Neural Information Processing Systems, 2023, 36: 51094 – 51112.
[28] WANG C Y, YEH I H, LIAO H Y. Yolov9: learning what you want to learn using programmable gradient information[C]// European Conference on Computer Vision. Cham: Springer Nature Switzerland, 2024: 1 – 21.
[29] WANG A, CHEN H, LIU L, et al. YOLO v10: real-time end-to-end object detection[J]. Advances in Neural Information Processing Systems, 2024, 37: 107984 – 108011.
[30] LI X, WANG W, HU X, et al. Generalized Focal Loss V2: learning reliable localization quality estimation for dense object detection[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021.
[31] LI X, LV C, WANG W, et al. Generalized Focal Loss: towards efficient representation learning for dense object detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022,45(3):3139 – 3153.
[32] SHI D. TransNeXt: robust foveal visual perception for vision transformers[C]//2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024.
[33] BODLA N, SINGH B, CHELLAPPA R, et al. Soft – NMS — improving object detection with one line of code[C]//2017 IEEE International Conference on Computer Vision (ICCV), 2017.

基于 HLNC－YOLO 的密集养殖下湖羊行为识别方法

冀荣华[1]　常宏瑞[1]　张所向[1]　刘中英[2]　吴中红[2]
（1. 中国农业大学信息与电气工程学院，北京 100083；2. 中国农业大学动物科学技术学院，北京 100093）

摘要：湖羊的行为识别有助于其集约化和智能化养殖。在高密度饲养条件下，羊只间相互遮挡严重，导致现有行为识别方法易发生漏检与误检。为此，提出了一种高低频信息聚合的湖羊行为识别模型（HLNC－YOLO），用于识别密集养殖场景下的湖羊行为。采集养殖场湖羊站立、躺卧、进食和饮水 4 种典型行为图像，构建湖羊行为识别数据集 HSBD（Hu sheep behavior dataset）。在 YOLO v8 主干网络中融入高低频信息聚合的 C2F－HLAtt 模块以感知被遮挡目标，引入辅助判断分支保留更多有效特征，采用基于定位框综合分数的负样本综合损失（Comprehensive score regression loss，CSLosss）降低次优框分数，提升被遮挡目标框的综合分数。在模型推理阶段使用综合分数衰减后处理算法（Soft comprehensive score non-maximal suppression，Soft－CS－NMS）筛选预测框。在 HSBD 数据集上进行测试，HLNC－YOLO 的平均精度均值（mAP@50）为 87.8%，内存占用量为 17.4 MB。较 YOLO v8、YOLO v9、YOLO v10 和 Faster R－CNN 平均精度均值分别提高 7.1、2.2、4.6、11 个百分点。研究表明，HLNC－YOLO 实现了密集养殖湖羊行为的准确识别，具有泛化能力，可为智慧养殖提供技术支持。

关键词：行为识别；YOLO；损失函数；注意力机制

基金项目：国家重点研发计划项目（2022YFD1301104）

（上接第 69 页）

[23] 程子文，杨德秋，刘萌萌，等. 基于离散元仿真的马铃薯收获机集薯装袋过程研究[J]. 农业机械学报，2024，55（增刊 2）：62－74.
CHENG Ziwen，YANG Deqiu，LIU Mengmeng，et al. Bagging process of potato harvester based on EDEM[J]. Transactions of the Chinese Society for Agricultural Machinery，2024，55（Supp. 2）：62－74.（in Chinese）

[24] 崔王斌，周宏平，许林云，等. 我国林果机械振动式采收装备及理论研究进展[J]. 世界林业研究，2023，36（6）：64－70.
CUI Wangbin，ZHOU Hongping，XU Linyun，et al. Research progress in mechanically vibratory harvesting equipment of forest fruit and its theory in China[J]. World Forestry Research，2023，36（6）：64－70.（in Chinese）

[25] 李琳，陈芳，梁学强，等. 林果机械化采收技术研究进展[J]. 农业工程，2023，13（8）：10－13.
LI Lin，CHEN Fang，LIANG Xueqiang，et al. Research progress on mechanized harvesting technology of forest and fruit[J]. Agricultural Engineering，2023，13（8）：10－13.（in Chinese）

[26] 王海滨，李志鹏，姜雪松，等. 基于槽型凸轮传动的蓝莓采摘机设计与试验[J]. 农业机械学报，2018，49（10）：80－91.
WANG Haibin，LI Zhipeng，JIANG Xuesong，et al. Design and experiment on blueberry picking machine based on groove cam drive[J]. Transactions of the Chinese Society for Agricultural Machinery，2018，49（10）：80－91.（in Chinese）

[27] 刘洪杰，李峻藤，杨欣，等. 手持摇枝式酸枣振动采摘装置设计与试验[J]. 农业机械学报，2024，55（9）：194－204，215.
LIU Hongjie，LI Junteng，YANG Xin，et al. Design and experiment of hand-held shaking branch type jujube vibration picker[J]. Transactions of the Chinese Society for Agricultural Machinery，2024，55（9）：194－204，215.（in Chinese）

[28] 徐潺，范艳，张成玉，等. 适用于山地手持式电动蓝莓采摘机的研制[J]. 农机化研究，2021，43（12）：133－137.
XU Chan，FAN Yan，ZHANG Chengyu，et al. Research and development of hand-held pure electric blueberry picker suitable for mountainous regions[J]. Journal of Agricultural Mechanization Research，2021，43（12）：133－137.（in Chinese）

[29] 张波，杨化伟，卢绪振，等. 蓝莓采摘期结合力测试与变化趋势的分析[J]. 农业装备与车辆工程，2021，59（9）：27－31.
ZHANG Bo，YANG Hawei，LU Xuzhen，et al. Binding force test and trend analysis of blueberry picking period[J]. Agricultural Equipment & Vehicle Engineering，2021，59（9）：27－31.（in Chinese）

[30] 陈嘉瑶，王英，梁冬泰，等. 小型化轴向振动式蓝莓采摘机设计与试验[J]. 机械设计，2021，38（4）：37－43.
CHEN Jiayao，WANG Ying，LIANG Dongtai，et al. Design and experiment of the miniaturized axial-vibrating blueberry picker[J]. Journal of Machine Design，2021，38（4）：37－43.（in Chinese）

[31] 赵恩龙，赵金辉，庄腾飞，等. 基于油茶枝果力学模型的振动脱果仿真与试验[J]. 农业机械学报，2025，56（10）：365－374.
ZHAO Enlong，ZHAO Jinhui，ZHUANG Tengfei，et al. Simulation analysis and experiment of vibration fruit removal based on mechanical model of *Camellia oleifera*[J]. Transactions of the Chinese Society for Agricultural Machinery，2025，56（10）：365－374.（in Chinese）

[32] 王业成，袁威，陈海涛，等. 便携式小浆果采收器[J]. 农业机械学报，2011，42（增刊）：181－183，180.
WANG Yecheng，YUAN Wei，CHEN Haitao，et al. Handy harvester for berry[J]. Transactions of the Chinese Society for Agricultural Machinery，2011，42（Supp.）：181－183，180.（in Chinese）