Method for detecting dead caged laying ducks based on infrared thermal imaging

Yu Yan, Qiaohua Wang, Weiguo Lin, Shucai Wang*, Yue Gu, Yifan Heng
(*College of Engineering, Huazhong Agricultural University, Wuhan 430070, China*)

Abstract: To accurately and efficiently detect dead caged laying ducks, thereby reducing reliance on manual inspection, this study proposes a method that integrates infrared thermography with deep learning technology. A lightweight object detection algorithm is developed, utilizing YOLO v8n as the baseline model. The backbone network is replaced with StarNet, which is based on "Star Operate". Additionally, the C2f-Star structure is designed by combining the Star Block from StarNet with the C2f module, and it is inserted into the Neck structure of the baseline model. Lightweight module L-SPPF replaces the SPPF module in the baseline model to enhance feature augmentation. Furthermore, a lightweight shared convolutional detection head, termed SCSB-Head, is introduced to reduce computational complexity. These improvements collectively form a lightweight object detection algorithm named SLSS-YOLO. Experimental results show that SLSS-YOLO achieves mAP@50%-95%, precision, and recall scores of 80.50%, 99.44%, and 98.46%, respectively. Compared to the baseline model, these metrics improve by 1%, 1.98%, and 0.26%, respectively. In terms of model size and detection speed, SLSS-YOLO has 1.44 M parameters and 4.6 G FLOPs, achieving an FPS rate of 134.9 f/s. This represents a reduction of 52.16% and 43.90% in parameters and FLOPs, respectively, while increasing FPS by 5.4 f/s compared to the baseline model. Moreover, an object tracking model is constructed using SLSS-YOLO and Hybrid-SORT. Tracking tests demonstrate that Hybrid-SORT achieves zero ID-Switches, with a detection speed of 10.9 ms/f. It outperforms Bot-SORT, ByteTrack, Deep OC-SORT, and OC-SORT in terms of tracking performance. Therefore, the proposed thermal infrared detection method can effectively identify and track dead ducks in complex cage environments, providing a reference for automated inspection in caged duck farms.
Keywords: caged laying duck, object detection algorithm, YOLO, infrared thermal imaging, dead poultry
DOI: 10.25165/j.ijabe.20241706.8314

Citation: Yan Y, Wang Q H, Lin W G, Wang S C, Gu Y, Heng Y F. Method for detecting dead caged laying ducks based on infrared thermal imaging. Int J Agric & Biol Eng, 2024; 17(6): 101–110.

1 Introduction

China hosts over 90% of the world's farmed laying ducks. Large-scale caged farming of laying ducks is a critical approach to enhancing production efficiency[1-3]. However, this method presents common issues: manual inspections are still required to monitor individual health, which can increase stress, trampling, and pecking behaviors detrimental to their well-being[4]. Additionally, if sick or dead poultry are not promptly identified and removed, they can severely contaminate the farm environment, leading to widespread infections[5-6]. Therefore, automating the rapid and accurate identification of dead ducks is essential to addressing the reliance on manual inspections in large-scale caged farming.

Advances in computer vision have deepened research areas such as behavior recognition and health monitoring in livestock animals[7-8]. Currently, object detection algorithms based on CNN and Transformer architectures are widely applied in livestock studies, showing significant effects[9-11]. Xiao et al.[12] proposed a DHSW-YOLO model to automatically monitor and warn of behavioral rhythms in ducks. Bist et al.[13] used the YOLOv5n-BFD model to detect leg disorders in broilers. Li et al.[14] employed the EMSC-DETR algorithm to locate and detect free-range chickens in complex environments. Sachin et al.[15] used the YOLOv5s-pecking model to detect feather pecking behavior in laying hens in non-cage settings. Additionally, infrared thermography is widely used in livestock detection[16-18]. Xie et al.[19] combined infrared thermography with an improved YOLO v5s-BiFPN to automatically detect body temperatures of pigs. Schreiter et al.[20] analyzed pixel data from infrared images of laying hens to assess feather damage. Wang et al.[21] used the YOLOv5 deep learning model to obtain temperature information from eyes and udders, enabling mastitis detection in dairy cows. Pacheco et al.[22] designed four deep learning models based on a CNN architecture to identify and evaluate the degree of heat stress in dairy cows.

Infrared thermography is a non-invasive monitoring technology[23]. For intensively farmed laying ducks, automatic inspection equipment must quickly and accurately identify dead ducks, unaffected by lighting conditions. This is an important technological means for advancing intelligent inspection in caged duck farming. Therefore, this study developed a lightweight object detection network, SLSS-YOLO, using infrared thermographic images. Combined with Hybrid-SORT, this study proposes a method for detecting and tracking dead ducks in caged duck farms.

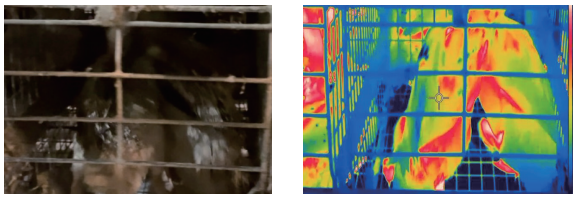
2 Materials and methods

2.1 Collection of infrared thermal images

The experimental data was collected at a welfare laying duck

Received date: 2023-04-26 **Accepted date:** 2024-10-16
Biographies: Yu Yan, PhD candidate, research interest: poultry farming robots and computer vision detection, Email: yu.yan@webmail.hzau.edu.cn; **Qiaohua Wang**, PhD, Professor, research interest: intelligent detection equipment for poultry eggs, Email: wqh@mail.hzau.edu.cn; **Weiguo Lin**, PhD, Associate Professor, research interest: digital intelligent detection of agricultural products, Email: linweiguo@mail.hzau.edu.cn; **Yue Gu**, PhD candidate, research interest: intelligent detection of poultry behavior, Email: yueegu@outlook.com; **Yifan Heng**, PhD candidate, research interest: intelligent detection of physiological information in poultry, Email: a1715622858@163.com.
***Corresponding author: Shucai Wang**, PhD, Professor, research interest: intelligent equipment for livestock and poultry products. Huazhong Agricultural University, Shizishan Street, Hongshan District, Wuhan 430070, Hubei Province. Tel: +86-13986165641, Email: wsc01@mail.hzau.edu.cn.

farm in Anlu City, Hubei Province. The subjects were ducks raised in stacked cages. Data collection occurred in May 2022, with temperatures in the duck house ranging from 26°C to 28°C. Visible light and thermal infrared images taken on-site are shown in Figure 1a and 1b. The visible light images of the cages appear dim with a complex background, whereas the pseudo-color thermal images are clearer. The experimental duck house was an H-shaped five-story facility. Each story housed 132 ducks. Each cage was 446 mm long, 416 mm wide, and 300 mm high. Most cages held two ducks, with a few holding only one.



a. Visible light image b. Thermal infrared image

Figure 1 Visible light camera and thermal infrared camera photography

The acquisition device was a FLIR T530 infrared thermal imager (FLIR Inc., Wilsonville, OR, USA). The infrared resolution was 320×240 pixels, with a wavelength range of 7.5 to 14.0 μm. The measurement temperature range was –20°C to 120°C, with an accuracy of ±2°C. The lens had a focal length of 17 mm (24°) and a thermal sensitivity of <30 mK. The thermal imager was mounted on a tripod beside the cages, positioned 600 mm from the cages and 1000 mm above the ground. Due to obstructions from feed troughs in the cage structure, the thermal imager was set at a 30° angle to the horizontal ground. The schematic diagram of image acquisition and the actual setup are shown in Figure 2.

Since the positions of dead ducks were not fixed, the tripod was manually moved to collect videos, simulating the inspection process. The video shooting process is shown in Figure 3. As shown in Figure 3, the feeding trough obscures part of the cage area. Therefore, the shooting angle in Figure 2 can reduce the obstruction of the feeding trough on dead ducks in the cage. To simulate the inspection process of the robot, this study performed horizontal movements of the infrared thermal camera according to the method shown in Figure 3.

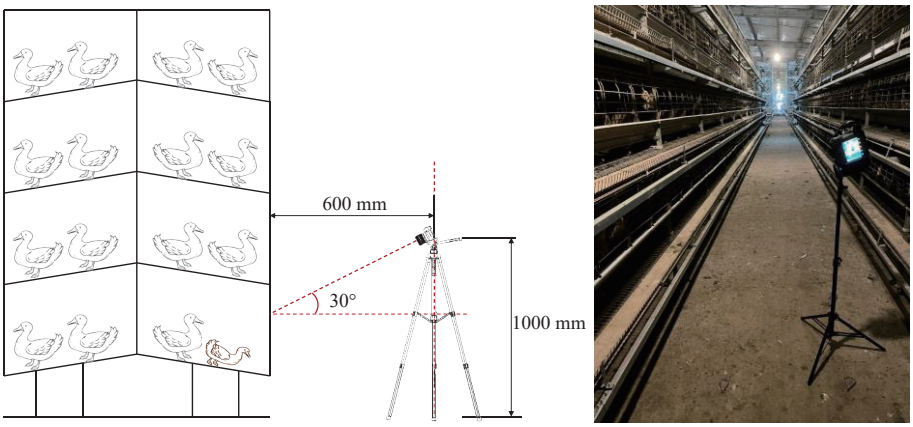
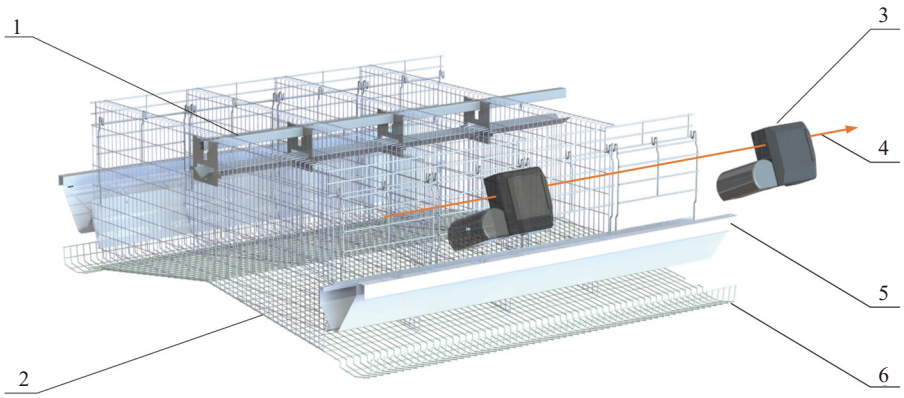


Figure 2 Image acquisition schematic diagram



Note: 1. Drinking equipment 2. The bottom of the cage net 3. Infrared thermal camera 4. Camera moving path 5. Feeding trough 6. Egg trough

Figure 3 Appearance diagram of duck cage and image acquisition process

2.2 Dataset creation

First, dynamic thermal infrared videos of dead ducks in cages were captured using a thermal infrared camera. Using video software, images were extracted at every fifth frame, resulting in 12 890 pictures. After removing images with no useful information or high overlap, 7848 images remained for the dataset. These images were manually annotated using LabelImg software (Tzutalin. LabelImg. Git code (2015). https://github.com/tzutalin/labelImg), labeling dead ducks as “dead_duck”. The annotations were formatted according to the Visual Object Class (VOC) dataset standard. The dataset was then randomly divided into test, training, and validation sets at a ratio of 1:8:1. The final distribution was 749 images for testing, 6061 for training, and 647 for validation.

3 Research methods

3.1 Baseline model: YOLO v8

YOLO v8[24] is the latest object detection algorithm released by the Ultralytics team. It comes in five versions, varying by model

size. YOLO v8n is the smallest model in this series. The architecture of YOLO v8 is primarily inherited from the team's previous one-stage algorithm, YOLO v5. Compared to YOLO v5, YOLO v8 uses a more lightweight C2f module instead of the original C3 module in the backbone network. It removes the original convolutional structure during upsampling in the feature pyramid network. In the detection head, it adopts an Anchor-Free mechanism, replacing the previous Anchor-Based system. Additionally, it introduces the Distribution Focal Loss (DFL) function to calculate regression losses. Overall, YOLO v8 better adapts to detecting irregularly shaped objects and reduces model complexity. Considering the requirements for later deployment on embedded devices, this study will use the smallest model, YOLO v8n, for detecting dead laying ducks in caged systems.

3.2 Improved model: SLSS-YOLO

The task of identifying dead laying ducks will be performed by an inspection robot based on an embedded development platform. However, YOLO v8 is designed for multi-category object detection. This study focuses on detecting dead ducks as a single target. Thus, redundant structures in the original model need improvement to reduce unnecessary computations and enhance overall recognition speed. Building upon the baseline model YOLO v8n, several modifications were made to the network structure. These include changes to the Backbone, Spatial Pyramid Pooling - Fast (SPPF), Neck, and Head components.

3.2.1 Backbone

To reduce computational load and enhance feature extraction, this study replaced the original CSPDarkNet backbone with StarNet[25]. StarNet is a new neural network model based on "element-wise multiplication" (Star Operate). "Star Operate" is a method that fuses features from different subspaces through element-wise multiplication. It resembles a multiplication operation denoted by a "star" symbol, demonstrating excellent performance and efficiency across multiple fields.

Compared to traditional convolutions or self-attention mechanisms, "Star Operate" exhibits higher computational efficiency. It maps inputs to a high-dimensional, nonlinear feature space, similar to polynomial kernel functions, thus improving model expressiveness. When integrated into neural networks and stacked in layers, each layer exponentially increases the hidden dimension complexity, allowing the model to achieve nearly infinite dimensions within a compact feature space.

As shown in Figure 7, the StarNet structure uses a hierarchical network design. It directly reduces resolution with convolutional layers and doubles the number of channels at each stage. Multiple Star Blocks repeat to extract features, with "Star Operate" used for feature fusion within each block.

3.2.2 Neck modifications

The C2f module is a crucial component of the Neck structure. Its design aims to extract and transform input data features through operations like feature transformation, branching, and fusion, generating more representative outputs. In the C2f module, Bottleneck structures reduce the number of channels in feature maps using 1×1 convolutional layers to decrease computation and memory consumption. Feature extraction follows via 3×3 convolutional layers, and the channel count is restored using another 1×1 convolutional layer.

To further reduce computational complexity and enhance efficiency, this study replaced the Bottleneck structures in the C2f module with Star Blocks, designing a new C2f variant called Star-C2f. This approach leverages the high computational efficiency of "Star Operate" to boost feature extraction capabilities within the C2f module. The redesigned Star-C2f module structure is illustrated in Figure 4.

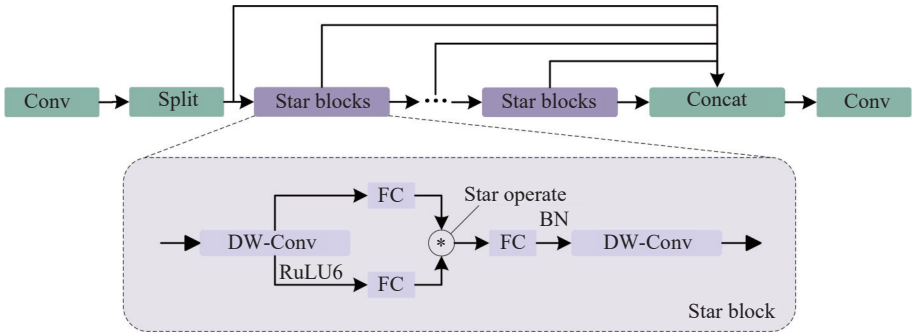


Figure 4 Structure diagram of Star-C2f module

3.2.3 SPPF modifications

SPPF is a feature fusion module in YOLO v8n that performs downsampling at different scales on the input feature maps through multiple max pooling operations. These downsampled feature maps, along with the original feature map, are concatenated (concat). The concatenated feature map is then processed by a convolutional layer (conv) to increase its dimensionality, producing the output feature map. However, SPPF does not distinguish between important and non-important regions of the feature map. When dealing with occluded objects, it captures too much local information, affecting model accuracy.

To address this issue, the SPPF structure was optimized by incorporating the Large Separable Kernel Attention (LSKA) module[26], creating an L-SPPF structure. LSKA is an improved version of the Large Kernel Attention (LKA) found in Visual Attention Network (VAN)[27]. LSKA decomposes the 2D convolutional kernel of a depthwise convolution into cascaded horizontal and vertical 1-D kernels. This decomposition allows for direct use of large-kernel depthwise convolutions in the attention module without additional blocks.

As shown in Figure 5, the LSKA module was placed after the concat operation in the SPPF structure. In the L-SPPF structure, the original SPPF achieves multiscale feature fusion through multiscale pooling. Meanwhile, the LSKA attention mechanism captures extensive contextual information from images using its large separable convolutional kernels and spatial dilated convolutions, generating richer feature representations. Combining these two approaches enhances the model's ability to extract features at different scales. Additionally, while LSKA improves accuracy, it does not significantly increase computational load or parameter count. This means that the overall model does not suffer from increased computational burden due to the introduction of this module.

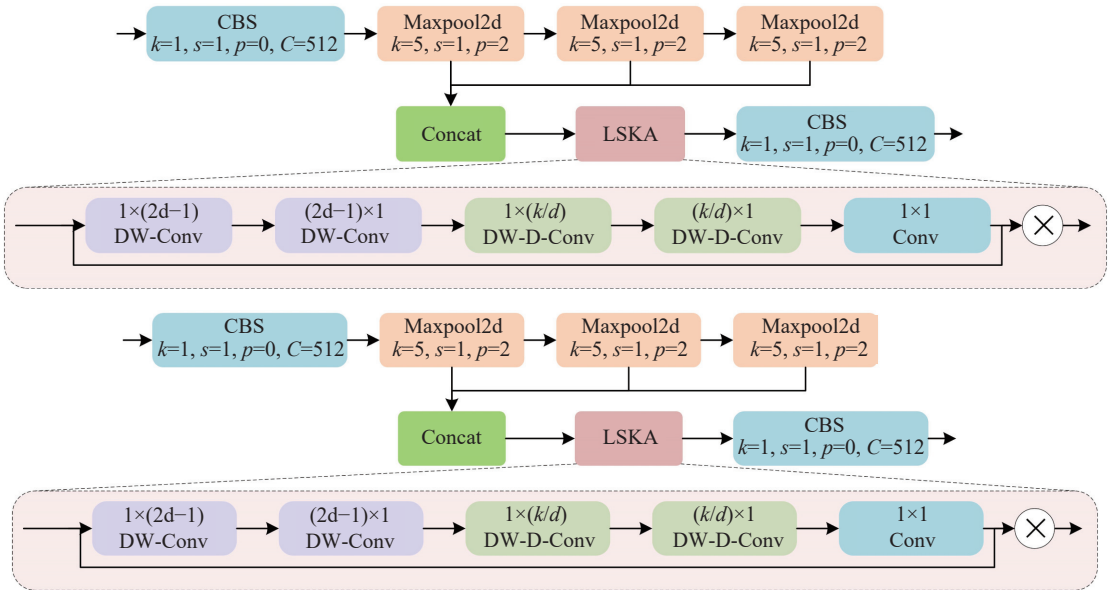


Figure 5 Structure diagram of L-SPPF

Therefore, the proposed L-SPPF structure enhances the model's overall feature attention capability without compromising inference speed, thereby improving performance in laying duck detection tasks.

3.2.4 Head modifications

The Head section of the YOLO v8 model includes a Decoupled-Head structure designed to accelerate convergence during training. This structure no longer uses a single convolution to share parameters for both classification and regression tasks. However, this design increases the number of convolutional operations, leading to higher computational demands. Consequently, under limited computational resources, YOLO v8 struggles to achieve optimal detection performance.

To address this issue, this study adopted a strategy inspired by RTMDet[28] and designed a Head structure called Shared Convolutional Separated Batch Normalization Head (SCSB-Head). As shown in Figure 6, the P3, P4, and P5 feature maps output from the Neck structure undergo 1×1 convolutions to adjust their channel sizes to a unified Chide. Subsequently, shared 3×3 convolution extracted features are then fed into separate Batch Normalization (BN) layers. Finally, a 1×1 convolution regression module predicts bounding box coordinate offsets. Further adjustments to feature scaling and target size are made by a Scale layer to locate dead laying ducks of different sizes. Simultaneously, a 1×1 convolution classification module predicts class probabilities. The weights of the convolutional layers in the classification and regression modules are independent, allowing the model to perform localization and

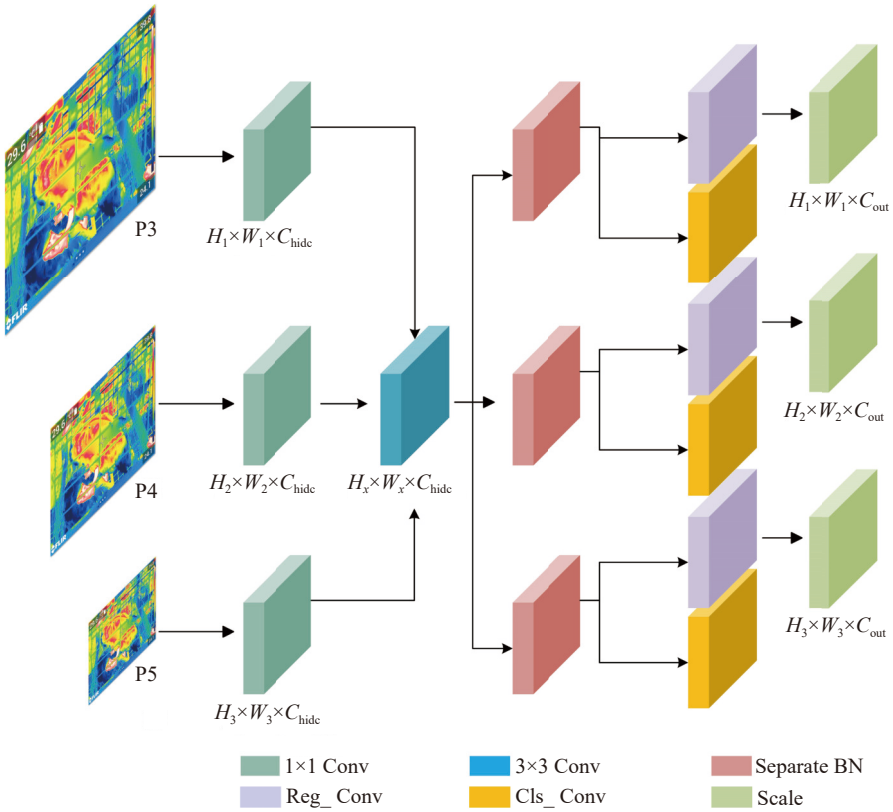


Figure 6 Structure diagram of SCSB-Head

classification tasks separately.

As shown in Figure 7, the modifications and reconstructions yield the new object detection network, SLSS-YOLO. From the figure, it is evident that the Backbone section has been updated from CSPDarkNet+SPPF to StarNet+L-SPPF. In the Neck structure, the C2f-Star module replaces the original C2f module. The Decoupled-Head structure has been redesigned into SCSB-Head.

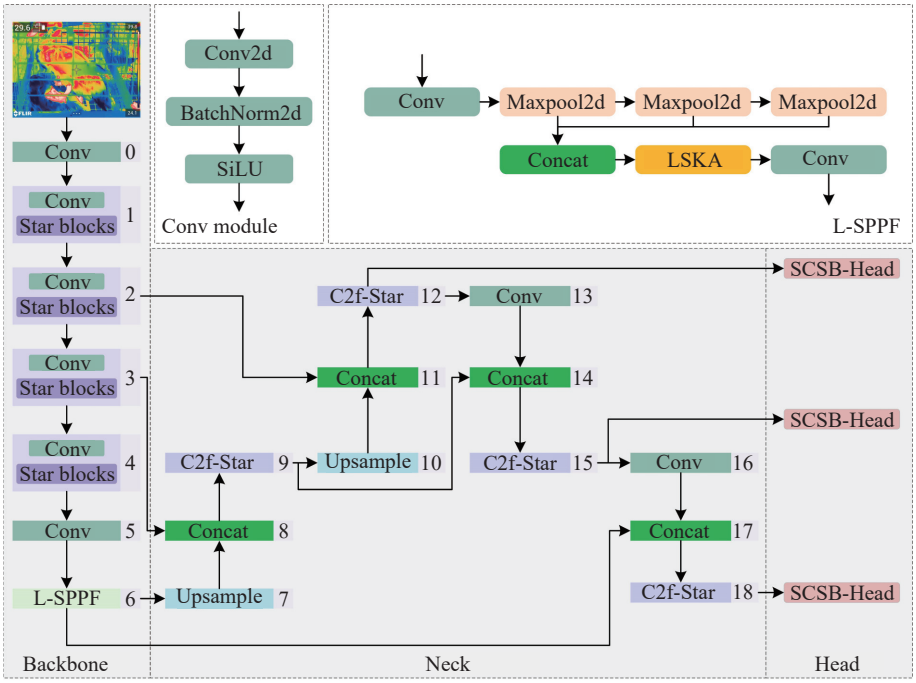


Figure 7 SLSS-YOLO model structure diagram

3.3 Object tracking algorithm

During inspections, multiple dead ducks are often present in the duck house. To distinguish and accurately track different individuals, this study combines the multi-object tracking model Hybrid-SORT[29] to track the status of dead ducks simultaneously. Hybrid-SORT addresses the shortcomings of traditional multi-object tracking algorithms in judging spatial and appearance information by introducing confidence state, height state, and velocity direction as weak cues. This model performs well when dealing with occlusion and clustering of objects.

In the confidence state cue, the Kalman filter state of each target is represented by Equation (1).

$$x = [u, v, s, c, r, \dot{u}, \dot{v}, \dot{s}, \dot{c}] \tag{1}$$

In Equation (1), x represents the Kalman filter state of trajectory confidence, while u and v denote the coordinates of the target's center. The parameter s represents the area of the target bounding box, and c represents the confidence of the trajectory. Additionally, r represents the aspect ratio of the target bounding box. The variables $\dot{u}$, $\dot{v}$, $\dot{s}$, and $\dot{c}$ represent the velocity components. The Kalman filter demonstrates effective continuous state estimation and modeling when targets are not heavily occluded. Hence, this method incorporates the trajectory confidence c and its velocity component $\dot{c}$ into the standard Kalman filter state equation.

However, when targets experience severe occlusion and clustering, the Kalman filter exhibits lagging behavior. To address this issue, linear prediction based on the trajectory history is employed when the confidence is low. The linear modeling equation is represented by Equation (2):

$$\hat{C}_{trk} = \begin{cases} C_{\text{trk}}^{t-1}, & C_{\text{trk}}^{t-2} = \text{None} \\ C_{\text{trk}}^{t-1} - (C_{\text{trk}}^{t-2} - C_{\text{trk}}^{t-1}), & \text{else} \end{cases} \tag{2}$$

In Equation (2), $\hat{C}_{\text{trk}}$ represents the confidence of the trajectory, and t represents the frame number. Therefore, the confidence cost formula for this model is represented by Equation (3):

$$C_{\text{Conf}} = \left| \hat{C}_{\text{trk}} - C_{\text{det}} \right| \tag{3}$$

where, C_{Conf} represents confidence cost, and C_{det} represents detection confidence.

In height state cue, the calculation disparity between Intersection over Union (IoU) and Height IoU (HIoU) is illustrated in Figure 8. The height of the target within the image recognition box across different frames to some extent reflects the depth information of the target, thereby compensating for the discrimination against strong cues.

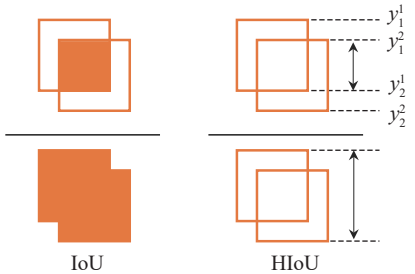


Figure 8 Computing method of IoU and HIoU

Therefore, the model integrates the two by introducing Height Modulated IoU (HMIoU), as shown in Equations (4)-(6).

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \tag{4}$$

$$\text{HIoU} = \frac{\min(y_2^1, y_2^2) - \max(y_1^1, y_1^2)}{\max(y_2^1, y_2^2) - \min(y_1^1, y_1^2)} \tag{5}$$

$$\text{HMIoU} = \text{HIoU} \cdot \text{IoU} \tag{6}$$

where, y_1^1, y_2^1, y_1^2, and y_2^2 respectively denote the coordinates of the

top and bottom boundaries of the two boxes.

In the velocity state cue, the velocity state of the target box's four corners is utilized instead of the velocity state of the original center point. This approach better expresses the direction and speed of the target's movement. The velocity cost function is illustrated by Equation (7).

$$C_{\mathrm{Vel}}=\sum_{\Delta t=1}^{3}\frac{(C_{\Delta t}^{\mathrm{lt}}+C_{\Delta t}^{\mathrm{rt}}+C_{\Delta t}^{\mathrm{lb}}+C_{\Delta t}^{\mathrm{rb}})}{4} \tag{7}$$

where, $C_{\Delta t}^{\mathrm{lt}}$, $C_{\Delta t}^{\mathrm{rt}}$, $C_{\Delta t}^{\mathrm{lb}}$, and $C_{\Delta t}^{\mathrm{rb}}$ respectively represent the velocity costs of the four vertices of the target box, while C_{Vel} denotes the average velocity cost of these four vertices. Thus, the total cost matrix is depicted by Equation (8):

$$C=C_{\mathrm{HMIoU}}+\lambda_1 C_{\mathrm{Vel}}+\lambda_2 C_{\mathrm{Conf}}+\lambda_3 C_{\mathrm{Appr}} \tag{8}$$

where, C_{Appr} represents the appearance cost, C_{HMIoU} denotes the height cost, and λ_1, λ_2, and λ_3 are the weighting parameters for each sub-cost function.

This study uses detection results for tracking, combining SLSS-YOLO with Hybrid-SORT to establish an object tracking model. The flowchart of the algorithm is shown in Figure 9.

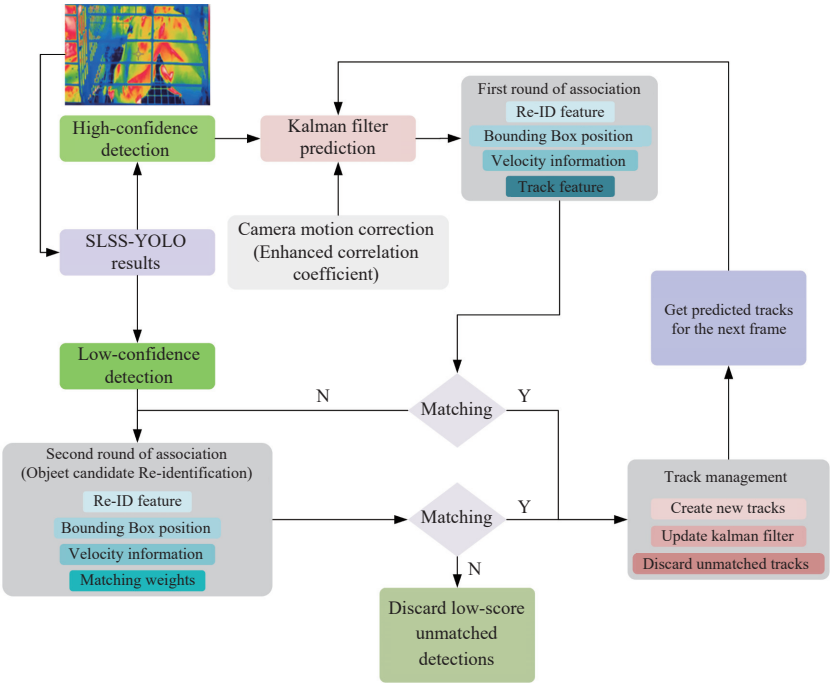


Figure 9 Target tracking model based on the combination of SLSS-YOLO and Hybrid-SORT

3.4 Evaluation metrics

1) Precision

Precision is the ratio of correctly predicted positive samples to all predicted positive samples. The calculation is shown in Equation (9):

$$\mathrm{Precision}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{9}$$

where, True Positives (TP) is the number of correctly detected positive samples; False Positives (FP) is the number of incorrectly detected positive samples.

2) Recall

The Recall is the ratio of correctly predicted positive samples to all actual positive samples. The calculation is shown in Equation (10).

$$\mathrm{Recall}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{10}$$

In the equation, TP (True Positives) is the number of correctly detected positive samples; FN (False Negatives) is the number of missed positive samples or incorrectly detected negative samples.

3) Mean Average Precision

Mean Average Precision (mAP) is a commonly used metric to evaluate the performance of object detection algorithms. mAP at an IoU of 50%-95% (mAP @50%-95%) indicates the average precision where the IoU threshold ranges from 50% to 95%, incrementing by 5%. It comprehensively evaluates the model's performance across multiple different overlap thresholds. mAP is the average of the Average Precision (AP) values at different IoU (Intersection over Union) thresholds, which can be calculated by Equation (11).

$$\mathrm{mAP}=\frac{1}{C}\sum_{C=1}^{C}\times\frac{1}{N}\sum_{n=1}^{N}\times\mathrm{AP}_{c,n} \tag{11}$$

where, C is the total number of classes, N is the number of IoU thresholds, and $\mathrm{AP}_{c,n}$ is the AP for class c at the n-th IoU threshold.

4 Results and analysis

4.1 Training environment and model parameter settings

The experimental conditions included a 64-bit Windows 10 operating system, an Intel(R) Core(TM) i5-10200H CPU @ 2.40 GHz, and an NVIDIA GTX1650Ti GPU. The PyTorch environment version was 1.10.1, and CUDA version was 11.3. During training, each model underwent 100 epochs with a batch size of 32. Mosaic data augmentation was applied for the first 10 epochs, and stochastic gradient descent (SGD) was used as the optimizer for model training.

4.2 Loss value comparison

During training, loss values were recorded at each iteration and written to the validation set. To evaluate the performance before and after model improvements, this study compared the loss values on the validation set. The changes in loss values are shown in Figure 10.

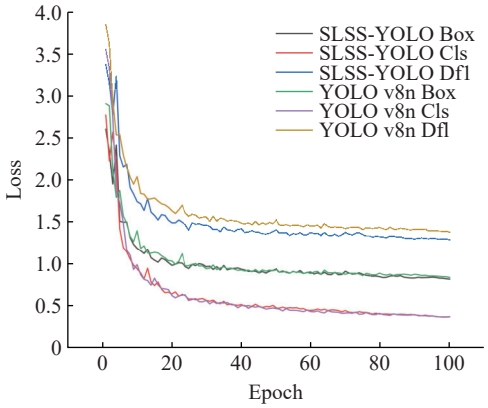


Figure 10 Change curves of loss value

The comparison includes Box Loss, Classification Loss, and Distribution Focal Loss, represented as Box, Cls, and Dfl, respectively. It is clear that the improved model shows almost identical convergence in Box Loss and Cls Loss. However, in Dfl Loss, the improved model demonstrates significantly faster convergence. This indicates that the improved model can quickly adjust its weight parameters from an initial state within a shorter training time. Thus, the improved model exhibits better learning capability and adaptability, positively impacting time costs and computational resources.

4.3 Ablation study

To verify the impact of each modification on the overall model training results, this study built four detection models using the aforementioned improvement strategies. These models were compared against the baseline model, YOLO v8n. The comparison results are listed in Table 1.

From the results in Table 1, Experiment 2 used StarNet as the backbone for YOLO v8n, reducing the number of parameters and FLOPs by 26.24% and 20.73%, respectively, compared to the baseline model. Performance metrics such as mAP@50%-95%, Precision, and Recall all improved. This network simplified the backbone using a hierarchical structure based on "Star operate", also enhancing efficiency in handling complex images.

Experiment 3 further modified the Neck section with the C2f-Star module, reducing parameters to 32.89% and FLOPs to 25.60% of the baseline model. Precision and Recall showed improvements over Experiment 2, due to the efficient feature extraction of the Star-C2f module.

Table 1 Ablation study results

NO.	StarNet	Star-C2f	SCSB-Head	L-SPPF	mAP_{50-95}/%	Precision/%	Recall/%	Parameters/M	FLOPs/G	FPS/f·s^{-1}
1	✗	✗	✗	✗	79.50	97.46	98.20	3.01	8.2	129.5
2	✓	✗	✗	✗	78.89	98.97	98.80	2.22	6.5	134.2
3	✓	✓	✗	✗	79.34	97.90	97.95	2.02	6.1	122.3
4	✓	✓	✓	✗	79.91	98.48	97.42	1.37	4.6	128.2
5	✓	✓	✓	✓	80.50	99.44	98.46	1.44	4.6	134.9

Note: ✓ indicates that the modification was applied to the baseline model, while ✗ indicates that the modification was not applied.

In Experiment 4, the model parameters and FLOPs changed significantly, decreasing by 54.49% and 43.90%, respectively, compared to the baseline model. Performance metrics such as mAP@50%-95% and Precision improved over Experiment 3. This indicates that the SCSB-Head structure can substantially reduce computational load and parameter count while maintaining overall detection accuracy. The proposed method's shared convolutional structures merged multiple convolutions, reducing total parameters by sharing convolutional parameters across detection heads at different scales. However, each scale's detection head used an independent BN layer. BN layers help mitigate internal covariate shift by normalizing the distribution of each mini-batch, enhancing model stability and generalization.

Experiment 5 utilized the LSKA module to modify the SPPF structure. Results showed parameter reductions of 52.16% and FLOPs reductions of 43.90% compared to the baseline model. Compared to Experiment 4, the parameter count slightly increased, but FLOPs remained largely unchanged. The model's mAP@50%-95% and Precision reached the highest levels among the four improvement strategies, with Recall only slightly below Experiment 2's 98.8%. This suggests that the L-SPPF structure, through large separable convolutional kernels and spatial dilated convolutions, generates richer feature representations and achieves multiscale feature fusion via multiscale pooling.

Thus, the proposed improvements in this study are effective.

4.4 Comparison with other object detection models

To further validate the performance of the improved model in detecting dead laying ducks in caged systems, this study conducted a comprehensive comparison with smaller versions of other object detection algorithms in the YOLO series. Using the same dataset, we trained the models on YOLO v4-tiny[30], YOLO v5n, YOLO v6n[31], YOLOv7-tiny[32], and YOLOv9-T[33]. The environment parameters were set identically for each model. The results are listed in Table 2.

Table 2 Comparison results of different models

Model	mAP_{50-95}/%	Precision P/%	Recall R/%	Parameters/M	FLOPs/G	FPS/f·s^{-1}
SLSS-YOLO	80.50	99.44	98.46	1.44	4.6	134.9
YOLO v4-tiny	77.60	95.26	94.53	5.92	16.1	83.7
YOLO v5n	78.72	96.46	97.92	2.51	7.2	128.1
YOLO v6n	79.29	96.44	97.10	4.24	11.9	114.8
YOLO v7-tiny	78.18	97.58	98.26	6.2	3.46	131.2
YOLO v9-T	78.26	97.46	96.41	2.00	7.8	59.7

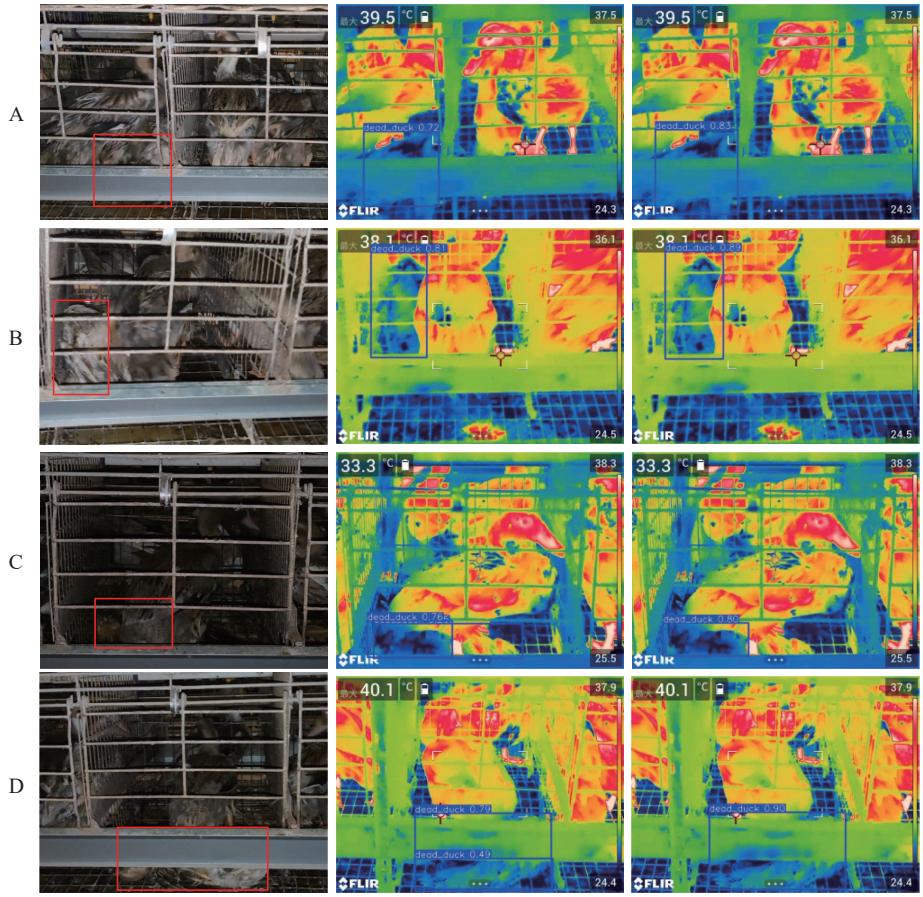
In terms of model size, SLSS-YOLO achieved the smallest number of Parameters and FLOPs among the small versions of YOLO models. In terms of performance, SLSS-YOLO's mAP_{50-95}, precision, and Recall were at the highest levels among the compared models. Therefore, SLSS-YOLO achieved the best detection performance with the smallest model size.

4.5 Comparison of detection effect for dead laying ducks in caged systems

To more significant compare the detection effectiveness before and after model improvements, images collected under the same conditions but not used for training were tested. As shown in Figure 11, four groups of comparison images were selected. In Groups A and B, both the YOLO v8n model and the SLSS-YOLO model detected dead ducks obscured by panels and live ducks. However, SLSS-YOLO exhibited higher confidence scores. In Group C, the dead duck was obstructed by live ducks and panels. Both models accurately identified the dead duck's location, but

YOLO v8n had lower confidence scores and produced multiple overlapping detection boxes. In Group D, where the dead duck was largely obscured by a panel, SLSS-YOLO accurately and completely detected the dead duck. YOLO v8n generated two detection boxes with lower confidence scores than SLSS-YOLO.

Across all four groups, when faced with detection tasks involving different positions and occlusions, the improved model SLSS-YOLO consistently demonstrated superior detection performance. Therefore, the improvement strategies in this study are effective.



a. Original visible light of image. The red box represents dead duck

b. Detection results YOLO v8n on the infrared thermal image

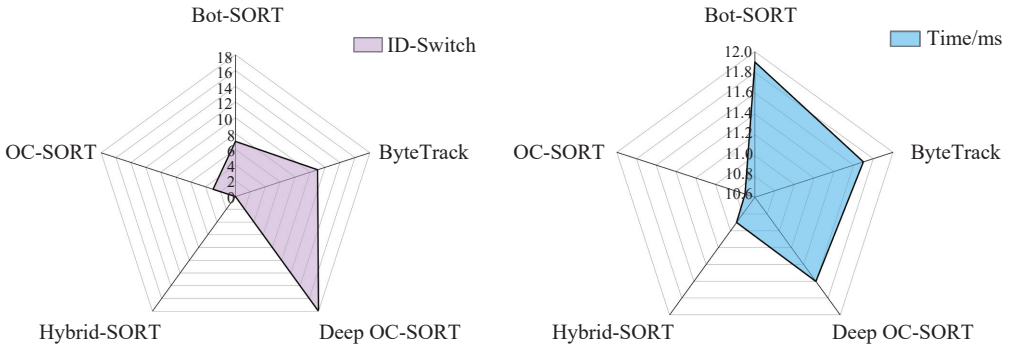
c. Detection results of SLSS-YOLO on the infrared thermal image

Figure 11 Inspection diagram of cage-reared laying ducks

4.6 Comparison of object tracking performance

This study used SLSS-YOLO combined with the Hybrid-SORT object tracking algorithm for tracking dead ducks. To compare the effectiveness of different tracking algorithms in identifying and tracking dead ducks, this study used randomly selected dead duck videos to validate and compare Bot-SORT[34], ByteTrack[35], Deep OC-SORT[36], and OC-SORT[37].

In object tracking, each object has a unique identifier (ID) to maintain consistency throughout the video sequence. ID-Switch, a common error, occurs when the tracking algorithm incorrectly assigns an ID from one object to another, leading to inconsistent tracking results and affecting overall performance. In this study, the number of IDs is key information for assessing the number of dead ducks. Therefore, Figure 12a shows the number of ID-Switches for five different tracking algorithms. Hybrid-SORT had zero ID-Switches, fewer than the other four algorithms.



a. Number of ID-Switch

b. Processing time for a single image

Figure 12 ID-Switch and processing time for a single image

Figure 12b illustrates the processing time per frame for the five algorithms, which includes preprocess time, inference time, and postprocess time. As shown, the processing time per frame for the OC-SORT algorithm was 10.7 ms. For Hybrid-SORT, it was 10.9 ms. The remaining three algorithms had processing times exceeding 11.6 ms. Although Hybrid-SORT had a slightly longer processing time than OC-SORT, the difference was minimal.

To visually compare the tracking detection results of the five algorithms, this study selected five sets of images from the detection outcomes, as shown in Figure 13. In the detection results, the correct ID for each dead duck was 0, and any other ID indicated an ID-Switch.

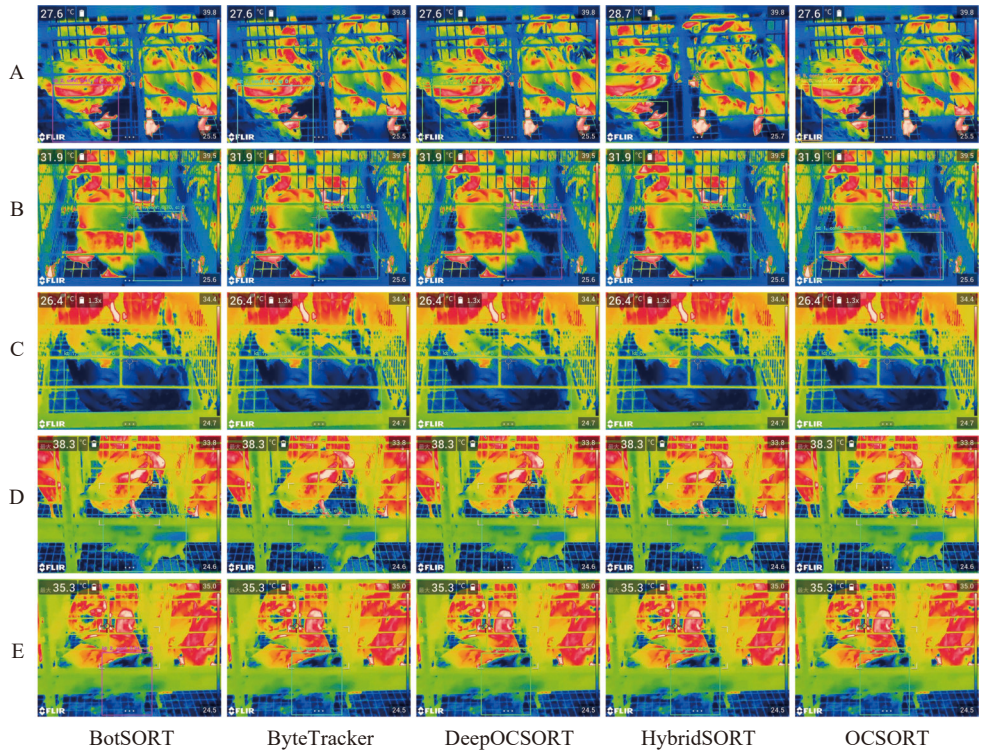


Figure 13 Comparison of object tracking performance

In Group A, the dead duck appears in the lower-left corner with only part of its body visible. It is evident that Deep OC-SORT and OC-SORT generated multiple detection boxes with obvious ID-Switches. Bot-SORT and ByteTrack failed to accurately track the dead duck, assigning IDs 2 and 7, respectively.

In Group B, the dead duck is located in the lower-right corner, partially obscured by another live duck. Bot-SORT, ByteTrack, and Deep OC-SORT all accurately detected the number and position of the dead duck but exhibited one ID-Switch each. OC-SORT's tracking results showed multiple detection boxes with noticeable ID-Switches.

In Group C, the dead duck is in the center, with only a few body parts obscured by a feeding trough board. All five algorithms accurately detected the position and number of the dead duck, but Bot-SORT, ByteTrack, and Deep OC-SORT each had one ID-Switch. Hybrid-SORT and OC-SORT correctly assigned the ID to the dead duck.

In Group D, the dead duck is partially obscured by a cage panel. All five algorithms accurately detected the position and number of the dead duck, but Bot-SORT, ByteTrack, and Deep OC-SORT each had one ID-Switch. Hybrid-SORT and OC-SORT correctly assigned the ID to the dead duck.

In Group E, the dead duck is partially obscured by a cage panel. All five algorithms accurately detected the position and number of the dead duck. Bot-SORT had two ID-Switches, ByteTrack had one ID-Switch, and Deep OC-SORT had four ID-Switches. Hybrid-SORT and OC-SORT did not exhibit any ID-Switches.

Therefore, combining the results in Figures 12 and 13, the tracking detection method using SLSS-YOLO and Hybrid-SORT outperforms other algorithms in tracking dead ducks in caged laying duck systems.

5 Conclusions

1) Addressing the issue of detecting dead ducks in caged laying duck systems, this study proposes a visual detection method based on infrared thermal imagery and deep learning. An infrared thermal camera was used as the image acquisition device for dead ducks, which can produce very clear images even under poor lighting conditions inside the cages. Additionally, this study mimicked the movement trajectory of inspection robots and selected a fixed angle to collect images of dead caged laying ducks, thereby avoiding the obstruction caused by the feeding trough.

2) In terms of detection methods, using YOLO v8n as the baseline model, this study developed a lightweight object detection algorithm, SLSS-YOLO. This algorithm employs StarNet as the backbone network. This study designed a C2f_Star structure by integrating the Star Block module with the C2f module and inserted it into the Neck structure of the baseline model. Additionally, this study constructed a lightweight SPPF structure, L-SPPF, using the LSKA module to enhance feature augmentation. Furthermore, this study designed a lightweight shared convolution detection head, SCSB-Head, to further reduce the model's computational load. In terms of detection performance, SLSS-YOLO improved the evaluation metrics mAP@50%-95%, Precision, and Recall by 1, 1.98, and 0.26 percentage points, respectively, compared to the baseline model. In terms of model size and detection speed, SLSS-YOLO reduced Parameters and FLOPs by 52.16% and 43.90%, respectively, compared to the baseline model. FPS increased by

5.4 frame/s. This lightweight object detection algorithm provides a reference for real-time detection of dead ducks by inspection robots.

3) To achieve tracking and detection of individual dead ducks, the object detection model SLSS-YOLO and the tracking model Hybrid-SORT were used to implement tracking recognition. Based on SLSS-YOLO, this study established five multi-object tracking models including Hybrid-SORT. In the comparative experiments, the proposed method achieved 0 ID-Switches and a detection speed of 10.9 ms/frame in test videos, demonstrating the best performance among the five algorithms. This method can accurately track and detect individual dead ducks, providing an effective approach for automated detection in intensive caged laying duck systems and offering technical references for analyzing individual dead ducks.

Acknowledgements

This study was supported by the Fundamental Research Funds for the Central Universities (Grant No. 2662022GXYJ004).

[References]

[1] Li Q X, Shao Z Y, Zhou W H, Su Q R, Wang Q H. MobileOne-YOLO: Improving the YOLOv7 network for the detection of unfertilized duck eggs and early duck embryo development-a novel approach. Computers and Electronics in Agriculture, 2023; 214: 108316.

[2] Huang C Y, Yu X C, Bao L, Mai L J, Li A. Ammonia negatively impacted egg production and altered the diversity of microbial communities in cage-reared Muscovy ducks. Poultry Science, 2024; 103: 103938.

[3] Liu H H, Qi J W, Yang Q L, Tang Q, Qi J J, Li Y Y, et al. Effects of cage and floor rearing systems on the metabolic components of the uropygial gland in ducks. Animals, 2022; 12(2): 214.

[4] Gu Y, Wang S C, Yan Y, Tang S J, Zhao S D, et al. Identification and analysis of emergency behavior of cage-reared laying ducks based on YoloV5. Agriculture, 2022; 12(4): 485.

[5] Wibisono F M, Wibisono F J, Effendi M H, Plumeriastuti H, Hidayatullah A R, Hartadi E B, Sofiana E D. A Review of Salmonellosis on Poultry Farms: Public Health Importance. Systematic Reviews in Pharmacy, 2020; 11(9): 481–486.

[6] Saraiva S, Saraiva C, Oliveira I, Stilwell G, Esteves A. Effects of age, weight, and housing system on prevalence of dead on arrival and carcass condemnation causes in laying hens. Poultry Science, 2021; 100: 100910.

[7] Jia W E, Qu H, Ma J, Xia Y T, Cui D J, Liu Y Y, et al. Adjacent age classification algorithm of yellow-feathered chickens based on multi-scale feature fusion. Computers and Electronics in Agriculture, 2022; 200: 107264.

[8] Fang C, Zhang T M, Zheng H K, Huang J D, Cuan K X. Pose estimation and behavior classification of broiler chickens based on deep neural networks. Computers and Electronics in Agriculture, 2021; 180: 105863,

[9] Jiang P Y, Ergu D, Liu F Y, Cai Y, Ma B. A Review of Yolo algorithm developments. Procedia Computer Science, 2022; 199(11): 1066–1073.

[10] Zhai S P, Shang D R, Wang S H, Dong S S. DF-SSD: An improved SSD object detection algorithm based on DenseNet and feature fusion. IEEE Access, 2020; 8: 24344–24357.

[11] Zhao Y A, Lv W Y, Xu S L, Wei J M, Wang G Z, Dang Q Q. Detrs beat yolos on real-time object detection. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA: IEEE, 2024; pp.16965–16974.

[12] Xiao D Q, Wang H D, Liu Y F, Li W G, Li H B. DHSW-YOLO: A duck flock daily behavior recognition model adaptable to bright and dark conditions. Computers and Electronics in Agriculture, 2024; 225: 109281.

[13] Bist R B, Yang X, Subedi S, Chai L. Automatic detection of bumblefoot in cage-free hens using computer vision technologies. Poultry Science, 2024; 103(7): 103780.

[14] Li X X, Cai M R, Tan X J, Yin C C, Chen W H, Liu Z, et al. An efficient transformer network for detecting multi-scale chicken in complex free-range farming environments via improved RT-DETR. Computers and Electronics in Agriculture, 2024; 224: 109160.

[15] Subedi S, Bist R, Yang X, Chai L. Tracking pecking behaviors and damages of cage-free laying hens with machine vision technologies. Computers and Electronics in Agriculture, 2023; 204: 107545.

[16] Bloch V, Barchilon N, Halachmi I, Druyan S. Automatic broiler temperature measuring by thermal camera. Biosystems Engineering, 2020; 199: 127–134.

[17] Oh S-I, Lee H S, Bui V N, Dao T D, Bui N., Le T D, et al. Dynamic variations in infrared skin temperature of weaned pigs experimentally inoculated with the African swine fever virus: A pilot study. Veterinary Sciences, 2021; 8(10): 223.

[18] Jorquera-Chavez M, Fuentes S, Dunshea F R, Warner R D, Poblete T, Morrison R S, et al. Remotely sensed imagery for early detection of respiratory disease in pigs: A pilot study. Animals, 2020; 10(3): 451.

[19] Xie Q J, Wu M R, Bao J, Zheng P, Liu W Y, Liu X F, et al. A deep learning-based detection method for pig body temperature using infrared thermography. Computers and Electronics in Agriculture, 2023; 213: 108200.

[20] Schreiter R, Freick M. Research Note: Is infrared thermography an appropriate method for early detection and objective quantification of plumage damage in white and brown feathered laying hens? Poultry Science, 2022; 101(9): 102022.

[21] Wang Y, Kang X, He Z, Feng Y, Liu G. Accurate detection of dairy cow mastitis with deep learning technology: A new and comprehensive detection method based on infrared thermal images. Animal, 2022; 16(10): 100646.

[22] Pacheco V M, Sousa R V, Sardinha E J S, Rodrigues A V S, Brown-Brandl T M, Martello L S. Deep learning-based model classifies thermal conditions in dairy cows using infrared thermography. Biosystems Engineering, 2022; 221: 154–163.

[23] Cai Z J, Cui J L, Yuan H B, Cheng M. Application and research progress of infrared thermography in temperature measurement of livestock and poultry animals: A review. Computers and Electronics in Agriculture, 2023; 205: 107586.

[24] Sohan M, Sai Ram T, Reddy C V R. A review on yolov8 and its advancements. In: Jacob I J, Piramuthu S, Falkowski-Gilski P. (Ed.). Singapore: Springer. International Conference on Data Intelligence and Cognitive Informatics. Springer, Singapore, 2024; pp.529–545.

[25] Ma X, Dai X Y, Bai Y, Wang Y Z, Fu Y. Rewrite the Stars. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA: IEEE, 2024; pp.5694–5703.

[26] Lau K W, Po L M, Rehman Y A U. Large separable kernel attention: Rethinking the large kernel attention design in CNN. Expert Systems with Applications, 2024; 236: 121352.

[27] Guo M-H, Lu C-Z, Liu Z-N, Cheng M-M, Hu S-M. Visual attention network. Computational Visual Media; 2023; 9(4): 733–752.

[28] Lyu C, Zhang W W, Huang H A, Zhou Y, Wang Y D, Liu Y Y, et al. Rtmdet: An empirical study of designing real-time object detectors. arXiv preprint arXiv: 2212.07784, 2022; doi: 10.48550/arXiv.2212.07784.

[29] Yang M Z, Han G Q, Yan B, Zhang W H, Qi J Q, Lu H C, et al. Hybrid-sort: Weak cues matter for online multi-object tracking. Proceedings of the AAAI Conference on Artificial Intelligence, 2024; 38(7): 6504–6512.

[30] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv: 2004.10934, 2020. doi: 10.48550/arXiv.2004.10934.

[31] Li C Y, Li L L, Jiang H L, Weng K H, Geng Y F, Li L, et al. YOLOv6: A single-stage object detection framework for industrial applications. arXiv preprint arXiv: 2209.02976, 2022. doi: 10.48550/arXiv.2209.02976.

[32] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada: IEEE, 2023; pp.7464–7475.

[33] Wang C Y, Yeh I H, Liao H Y M. Yolov9: Learning what you want to learn using programmable gradient information. arXiv preprint arXiv: 2402.13616, 2024. doi: 10.48550/arXiv.2402.13616.

[34] Aharon N, Orfaig R, Bobrovsky B-Z. BoT-SORT: Robust associations multi-pedestrian tracking. arXiv: 2206.14651, 2022; doi: 10.48550/arXiv.2206.14651.

[35] Zhang Y F, Sun P Z, Jiang Y, Yu D D, Weng F C, Yuan Z H, et al. Bytetrack: Multi-object tracking by associating every detection box. In: Computer Vision - ECCV 2022. ECCV 2022, Cham: Springer, 2022; doi: 10.1007/978-3-031-20047-2_1.

[36] Maggiolino G, Ahmad A, Cao J K, Kitani K. Deep OC-SORT: Multi-pedestrian tracking by adaptive re-identification. In: 2023 IEEE International Conference on Image Processing (ICIP), Kuala Lumpur, Malaysia: IEEE, 2023; pp.3025–3029.

[37] Cao J K, Pang J M, Weng X S, Khirodkar R, Kitani K. Observation-centric sort: Rethinking sort for robust multi-object tracking. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada: IEEE, 2023; pp.9686–9696.