Detection of Herd Pigs Based on Improved YOLOv5s Model

Jianquan LI[1], Xiao WU[2], Yuanlin NING[3], Ying YANG[4*], Gang LIU[5], Yang MI[6]
College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China[1,2,3,4,5,6]
Key Laboratory of Modern Precision Agriculture System Integration Research, Ministry of Education, Beijing 100083, China[5]
Key Laboratory of Agricultural Information Acquisition Technology, Ministry of Agriculture,
College of Information and Electrical Engineering, China Agricultural University, Beijing, China[5]

***Abstract*—Fast and accurate detection technology for individual pigs raised in herds is crucial for subsequent research on counting and disease surveillance. In this paper, we propose an improved lightweight object detection method based on YOLOv5s to improve the speed and accuracy of detection of herd-raised pigs in real-world and complex environments. Specifically, we first introduce a lightweight feature extraction module called C3S, then replace the original large object detection layer with a small object detection layer at the output (head) of YOLOv5s. Finally, we propose a dual adaptive weighted PAN structure to compensate for the information loss of feature map at the neck of YOLOv5s caused by down sampling. Experiments show that our method has an accuracy rate of 95.2%, a recall rate of 89.1%, a mean Average Precision (mAP) of 95.3%, a model parameter number of 3.64M, a detection speed of 154 frames per second, and a model layer count of 183 layers. Comparing with the original YOLOv5s model and the current state-of-the-art object detection models, our proposed method achieves the best results in terms of mAP and detection speed.**

Keywords*—Pig; deep learning; computer vision; object detection

I. INTRODUCTION

According to the data of the National Bureau of Statistics 0, the total meat production in China in 2022 was 92.27 million tons, of which the pork production was 55.41 million tons, accounting for about 60.05%. It can be seen that pig farming has become a pillar industry in China's livestock industry, and with the continuous expansion of pig farming, there is a growing demand for intelligent farming technologies to improve efficiency and productivity. Currently, the daily monitoring of pigs in herds is mainly performed by humans, which is subjective and time-consuming. Computer vision technologies such as object detection can realize automatic monitoring, which is more efficient and timely.

In recent years, many studies have investigated pig object detection. However, to obtain high detection accuracy in complex scenes such as piggery requires the deployment of large models, resulting in low detection speed and an inability to meet the real-time requirements. Furthermore, pigs tend to pile up and gather together, leading to severe occlusion and adhesion in images. In addition, pigs in images often appear as small objects occupying few pixels, which makes it more difficult to extract effective features and recognize. In summary, the main challenges faced by the current pig detection based on computer vision are:

- Detection speed and accuracy cannot be balanced, that means higher detection accuracy need bigger model size with lower detection speed, and it is difficult to meet the demand of high accuracy and real-time detection simultaneously.
- The heavily occluded nature of pigs in real farming scenarios presents a challenge for increasing detection accuracy.
- Detecting small objects such as pigs that occupy few pixels in the image is difficult and often leads to missed detections.

To address the above problems, we propose an improved object detection algorithm based on YOLOv5s for group-farmed pig. The main contributions of this paper are summarized as follows:

- A lightweight module is introduced for achieving better performance and faster speed simultaneously. The C3 module in YOLOv5s with more branches and deeper model layers has a large number of parameters and computation, which affects the model detection speed. Therefore, a more lightweight C3S module is used instead of C3 module to improve the detection speed without reducing the accuracy.
- A small object detection branch is added to the detection layer. Since small objects account for a substantial part of the dataset, and the characteristics of small objects are difficult to detect. A small object detection branch is added instead of the large object detection layer to improve the detection capability of the model for small objects.
- A dual adaptive weighted PAN structure is proposed to enhance the feature extraction ability of the neck. In view of the complexity of the real farming environment, the dual adaptive weighted PAN structure can extract more feature for object detection and thus improve the detection accuracy of the model.

II. RELATED WORKS

In recent years, a series of achievements have been made in the field of individual pig detection based on deep learning.

The detection algorithms mainly consist of one-stage and two-stage, with the one-stage representative algorithms including the YOLO series [2], SSD [5], FOCS [6], etc., and the two-stage representative algorithms including RCNN [7], Fast-RCNN [8], Faster-RCNN [9], etc.

In terms of one-stage algorithms, Yan et al. [10] improved the detection accuracy without introducing additional computation by combining Tiny-YOLO with feature pyramid attention, achieving an accuracy of 85.85% on detecting pigs in group breeding. Shen et al. [11] used the YOLOv3 and FPN algorithm for detecting piglets, achieving a detection accuracy of 93.84%. Fang et al. [12] improved the CenterNet by using MobileNet as the feature extraction network to reduce the number of parameters and increase the computation speed. By introducing a feature pyramid structure to enhance the feature extraction ability, they achieved an mAP of 94.3%. Seo et al. [13] reduced the computational workload of 3×3 convolutions in YOLOv4 to achieve fast detection of individual pigs and improved accuracy through the generation of a three-channel composite image using simple image preprocessing techniques. Ahn et al. [14] combined the test results of two YOLOv4 models at the bounding-box level to increase the pig detection accuracy from 79.93% to 94.33%. These one-stage object detection algorithms have achieved satisfactory detection accuracy in scenarios with lower pig density and less occlusion and adhesion. However, in practical applications, they may not accurately reflect the desired performance. In real breeding conditions, there is still room for improvement in striking a balance between detection accuracy and speed.

In terms of second-stage algorithms, Riekert et al. [15] combined NAS (Neural Architecture Search) with the Faster-RCNN to detect the posture and position of pigs, achieving an average detection accuracy of 80.2%. Li et al. [16] used ResNet101 combined with the FPN algorithm as the backbone network and trained Mask R-CNN with transfer learning to detect pig crawling behavior, achieving a detection accuracy of 94.5%. However, their detection speed does not satisfy the real-time detection requirements, and their large model size makes them difficult to deploy on embedded devices.

III. Datasets and Proposed Method

A. Datasets

The data in this study were collected from Tianpengxingwang pig breeding farm in Shunyi, Beijing in November 2019. A Hikvision camera was fixed above the pigsty at an oblique angle to cover the entire pig pen. The video was recorded in MP4 format with a resolution of 1920×1080 and a frame rate of 30 frames per second. A frame was extracted from the collected video every two minutes or every 3600 frames, resulting in a total of 500 images.

To enrich the background and shooting angles of the dataset, a publicly available group-feeding pig dataset provided by iFlytek was added. This portion of the dataset contains a total of 920 images with a resolution of 1920 × 1080, a bit depth of 24 bits, and 3-channel RGB color images. The two parts of the dataset contain 1,420 images and 43,592 pigs in total. The shooting angle is from a top-down perspective, and the shooting time includes both day and night scenes. Table I provides a statistical summary of pig density in the dataset used in this study. We augmented the images in the training set online, including HSV transformation, horizontal flipping, translation, proportional scaling, and Mosaic augmentation.

TABLE I. A Statistical Summary of Pig Densities

Number of individual pigs in a single image	Number of images
13-23	287
24-34	675
35-45	360
46-62	98

B. Improved YOLOv5s

In order to strike a balance between detection speed and accuracy, in this study, we consider adopting a one-stage approach as the base model. YOLOv5 [17] is a one-stage object detection model and has been improved on the basis of YOLOv4 [18], with the characteristics of small size, fast detection speed, and easy deployment. The improvement points that greatly improve its speed and accuracy mainly include the following four aspects:

- Input: Mosaic data augmentation, adaptive anchor box calculation and adaptive image scaling.
- Backbone: CSP structures, Focus structure.
- Neck: The Path Aggregation Network (PAN) [19] and Feature Pyramid Network (FPN) [20] structures are added between the backbone and head layers as a neck network.
- Head: Loss function CIoU-Loss [21] during train, IOU [22] during prediction and Non-maximum suppression (NMS) for prediction box screening.

There are four versions of YOLOv5, which are YOLOv5s, YOLOv5m, YOLOv5l and YOLOv5x, with the network depth and width increasing progressively. While the larger versions of YOLOv5 have higher detection accuracy, they also have a larger number of parameters and computation, making them less suitable for real-time detection scenarios. As a result, we have chosen to use YOLOv5s as the base network, as it has the smallest number of parameters among the four versions and can provide a reasonable trade-off between detection accuracy and computational efficiency. However, it still cannot fully meet the demands of detection of herd pigs, which require faster and more accurate object detection algorithms.

To address the challenges posed by severe occlusion and aggregation of pigs in group-raised pig farming scenarios and the requirement for real-time detection, we propose an improved version that significantly enhances its performance. Fig. 1 shows the structure details of improved YOLOv5s. By replacing C3 with C3S module, the number of parameters and calculation amount are greatly reduced, and the detection speed is accelerated while the accuracy remains unchanged. We also add a small object layer to improve the ability of the network to detect small objects. Finally, a dual adaptive weighted PAN structure is proposed to enhance the feature extraction ability and further improve its detection accuracy.

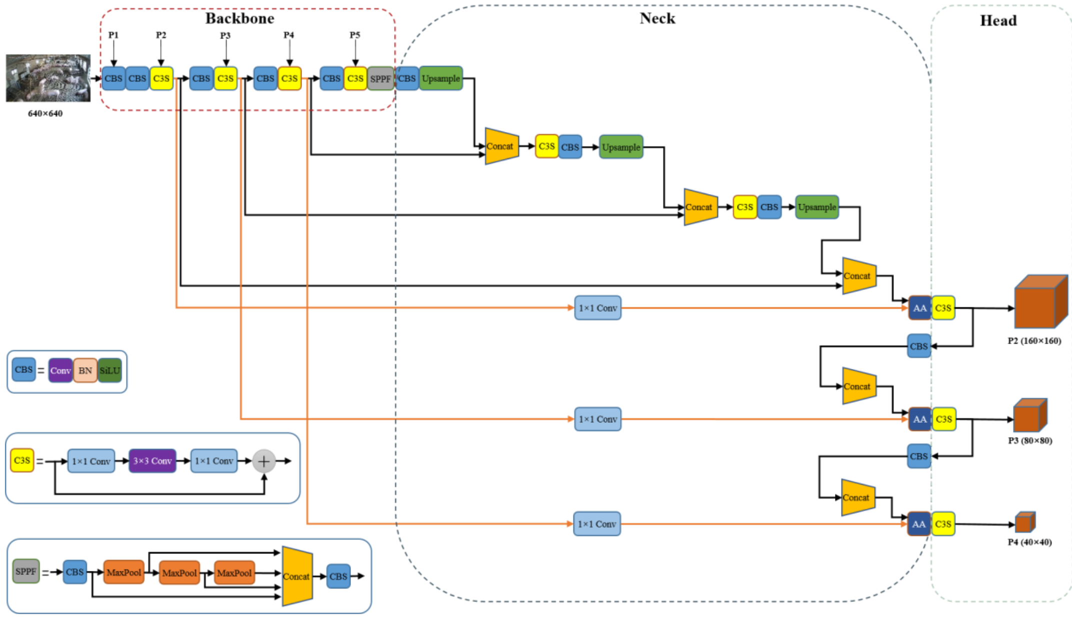


Fig. 1. The structure of improved YOLOv5s.

C. C3S Module

C3 module consists of convolutional layers (Conv), batch normalization layers (BN), the SiLU activation function, addition operations (add), and feature map concatenation along the channel dimension (Concat). The depth factor controls the number of BottleNeck modules in the structure and can be adjusted to control the model's depth. While the C3 module improves detection performance, it can also result in a deep model that reduces inference speed and increases computational cost and parameter count.

To improve the efficiency of the model, we introduced a lightweight convolutional module called C3S, which replaced the original C3 module. In Fig. 1, the C3S module is composed of 1x1 convolutions, 3x3 convolution, and residual structure to enhance the model's expressive power and feature extraction ability. Specifically, the 1x1 convolution can not only reduce the channel dimension but also promote inter-channel information exchange. Therefore, we first use 1x1 convolutions to reduce the input feature map's channel number by half, aiming to decrease the parameter count and computation cost. Next, we incorporate 3x3 convolutions to strengthen the feature extraction ability, increasing the feature map's channel number to twice the current number and subsequently using another 1x1 convolution to perform cross-channel information integration. Finally, we introduce residual structure to prevent gradient vanishing or explosion during model training. To further improve efficiency, we also decrease the channel number of the C3S module to 3/4 of the original by controlling the width factor.

D. Small Object Detection Layer

The input image resolution is 640×640, and then the downsampling operation is used by convolution with a stride of two, resulting in output feature maps with half the width and height of the input feature map. P1, P2, P3, P4, and P5 denote feature maps obtained via convolutional layers with downsampling steps of 1, 2, 3, 4, and 5, respectively, resulting in resolutions of 320×320, 160×160, 80×80, 40×40, and 20×20. As shown in Fig. 2, the head of YOLOv5s model consists of three detection layers, which take input feature maps of different resolutions (P3, P4, and P5). The P3 feature map, with the lowest resolution, is used to detect small objects, while the P4 and P5 feature maps are used to detect medium and large objects, respectively. In our dataset, small objects account for a large proportion of the total. Therefore, we replace the original large object detection layer in the head with a smaller one (as indicated by the red solid line) to enhance the model's ability to detect small objects in the images. The removed module is indicated by the light green dashed line.

Directly upsampling the feature map results in four times computational cost increase, which can negatively impact the inference speed. To address this issue, we first reduce the dimension of the input feature map before upsampling, which can partially alleviate the increase in parameters and computational cost. This approach improves the model's inference speed compared to direct upsampling.

$$Caculated = B\times O\times[C\times(\frac{H-K+P_h}{S}+1)\times(\frac{W-K+P_w}{S}+1)\times(2\times K\times K-1) \\ +(C-1)\times(\frac{H-K+P_h}{S}+1)\times(\frac{W-K+P_w}{S}+1) \tag{1}$$

Eq. (1) represents the calculation formula for the convolutional computation. In this equation, $Caculated$ represents the convolutional computation, B represents the batch size used during training, O represents the number of input feature map channels, C represents the number of output feature map channels, H and W represent the height and width of the input feature map, P_w and P_h represent the number of pixels padded in the height and width directions, respectively, S represents the stride of the convolutional kernel and K represents the size of the convolutional kernel (H and W are much larger than K , S , P_w and P_h).

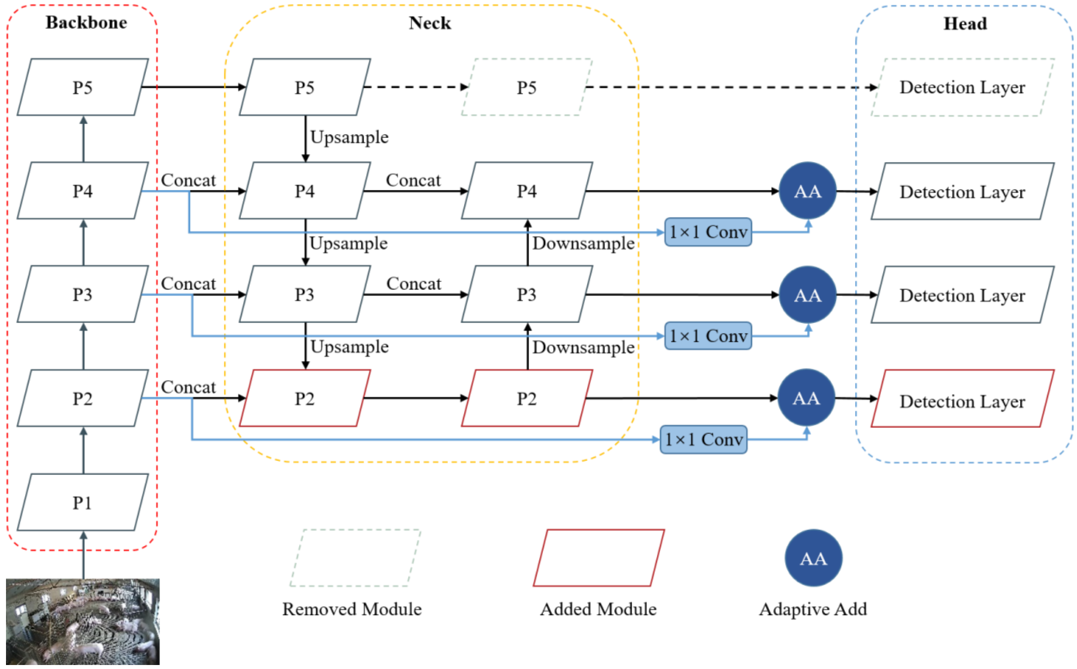


Fig. 2. Sketch of the model structure after adding the small object detection layer and dual adaptive weighted PAN structure.

E. *Dual Adaptive Weighted PAN Structure*

A dual adaptive weighted PAN structure was proposed to enhance the feature extraction ability of the neck. The blue part in Fig. 2 represents the dual adaptive weighted PAN structure, which consists of 1x1 convolutions and adaptive addition (AA) operations. In the YOLOv5s model, the features extracted by the backbone are fed into the neck for feature fusion. The neck performs multiple downsampling and upsampling operations to generate feature maps of different sizes for detecting objects of different sizes. However, some feature information is inevitably lost during the downsampling process. To address this issue, we reuse the original features extracted by the backbone and adjust their channel numbers using 1x1 convolutions to match the channel numbers of the large, medium, and small object feature maps output by the neck. We then perform adaptive addition between the adjusted feature maps and the object feature maps. Since the importance of backbone features and neck features may not be the same, direct addition may assume equal importance. Therefore, we define this addition operation as adaptive weighted addition, where a learnable weight is used to adjust the importance of the two types of features. We train the model using backpropagation and gradient descent to update the weight until convergence is reached.

$$X_{out} = w \times X_{input1} + (1-w) \times X_{input2} \tag{2}$$

Eq. (2) represents the adaptive add operation, where X_{out} represents the output feature map, w represents the weight, X_{input1} represents input feature map A, and X_{input2} represents input feature map B.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. *Experimental Parameters Setting*

The experiments in this study were conducted in Linux Ubuntu 18.0.4 environment (CPU: Inter Core i9 10900K, GPU: Nvidia GeForce RTX3060×2, RAM: 64G), and the deep learning framework was Pytorch 1.9.0. See Table II for other parameter settings. The experiments in this paper were conducted using the same experimental configuration.

TABLE II. PARAMETERS SETTING IN THIS PAPER

Configuration	Value
Optimizer	SGD
Learning Rate	0.01
Momentum	0.937
Weight Decay	0.0005
Batch Size	8
Training Epochs	200

B. *Evaluation Metrics*

The evaluation metrics used in this paper are precision, recall, mAP, model depth, parameter count, computational complexity, F1 score, and Frames Per Second (FPS). Precision represents the proportion of true positive predictions among all positive predictions made by the model, while recall represents the proportion of true positive predictions among all actual positive instances. The mAP is related to both precision and recall, with a higher mAP indicating a higher average detection accuracy of the model. The model's parameter count, computational complexity, and depth affect the inference speed of the model, while FPS measures the number of images the model can process per second, indicating the computational speed of the model. The F1 score is the harmonic mean of precision and recall and is used to measure the overall performance of the model. The equations for calculating these evaluation metrics are as follows:

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$AP = \int_0^1 PdR \tag{5}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{6}$$

$$F1 = \frac{2 \times (Presicion \times Recall)}{Presicion + Recall} \tag{7}$$

In the above equations, N represents the total number of classes and TP , FP and FN represent the number of true positive predictions, false positive predictions and false negative predictions, respectively. P and R are abbreviations for precision and recall, respectively.

C. Impact of Different Detection Layers on Model Performance

To explore the impact of varying numbers of detection layers and input feature map resolutions on model performance, we performed three sets of comparative experiments, and the results are presented in Table III. Experiment 1 used YOLOv5's original detection layers ([P3, P4, P5]). Experiment 2 added a small object detection layer with input resolution P2 ([P2, P3, P4, P5]). Compared to Experiment 1, although the precision decreased by 0.4%, the recall and mAP increased by 1.7% and 0.7%, respectively, in Experiment 2. Therefore, the overall performance of the model with the added small object detection layer was superior to that of the original YOLOv5 model. Experiment 3 removed the large object detection layer with input resolution P5 ([P2, P3, P4]). Compared to Experiment 2, the precision, recall, and mAP of the model further improved in Experiment 3, with precision increasing by 0.4%, recall increasing by 0.7% and mAP increasing by 0.5%.

In summary, the performance of the model was improved by adding a small object detection layer, as the feature map with input resolution P2 contained more information about small objects, making it easier to detect them. Furthermore, removing the large object detection layer with input resolution P5 led to further improvements in the model's performance. We believe this is because the number of large objects in our dataset was relatively small, resulting in fewer positive samples allocated to the large object detection layer during training. As a result, the parameters of the large object detection layer were difficult to optimize, making it challenging to accurately predict the presence of large objects, which ultimately affected the overall detection accuracy.

TABLE III. IMPACT OF DIFFERENT DETECTION LAYERS ON MODEL PERFORMANCE

Experiment Number	Input of detection layer Precision(%)	Precision (%)	Recall (%)	mAP (%)
1	[P3，P4，P5][a]	94.8	85.3	93.3
2	[P2，P3，P4，P5]	94.4	87	94
3	[P2，P3，P4]	**94.8**	**87.7**	**94.5**

[a] The input of the detection layer [P3, P4, P5] indicates that there are 3 detection layers, and the input resolutions are 8, 16, and 32 times downsampled from the input image, respectively.

D. Comparison with Different Object Detection Models

To verify the superiority of the improved YOLOv5s model proposed in this paper for individual pig detection in group feeding, we compared its performance with the other five common object detection models, such as Faster-RCNN [9], CenterNet [23], YOLOv3 [4], YOLOv4 [18], and YOLOX [24]. The experiments were conducted using the same experimental configuration. Fig. 3 shows the training accuracy curves for the six models, with the horizontal axis representing Epoch and the vertical axis representing mAP values. It can be seen that our method achieved the highest accuracy during the training phase.

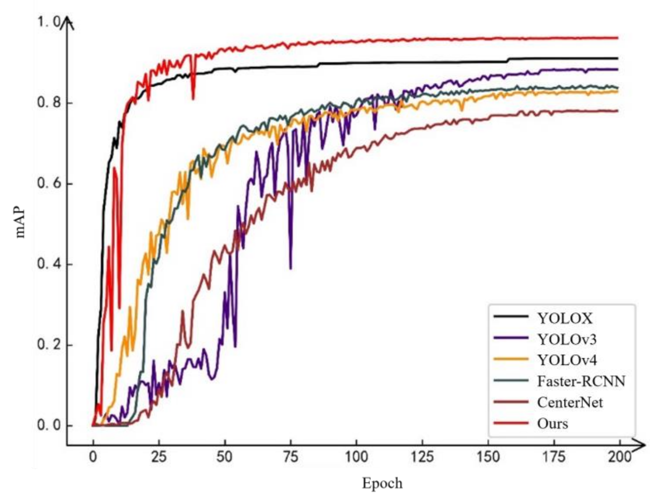


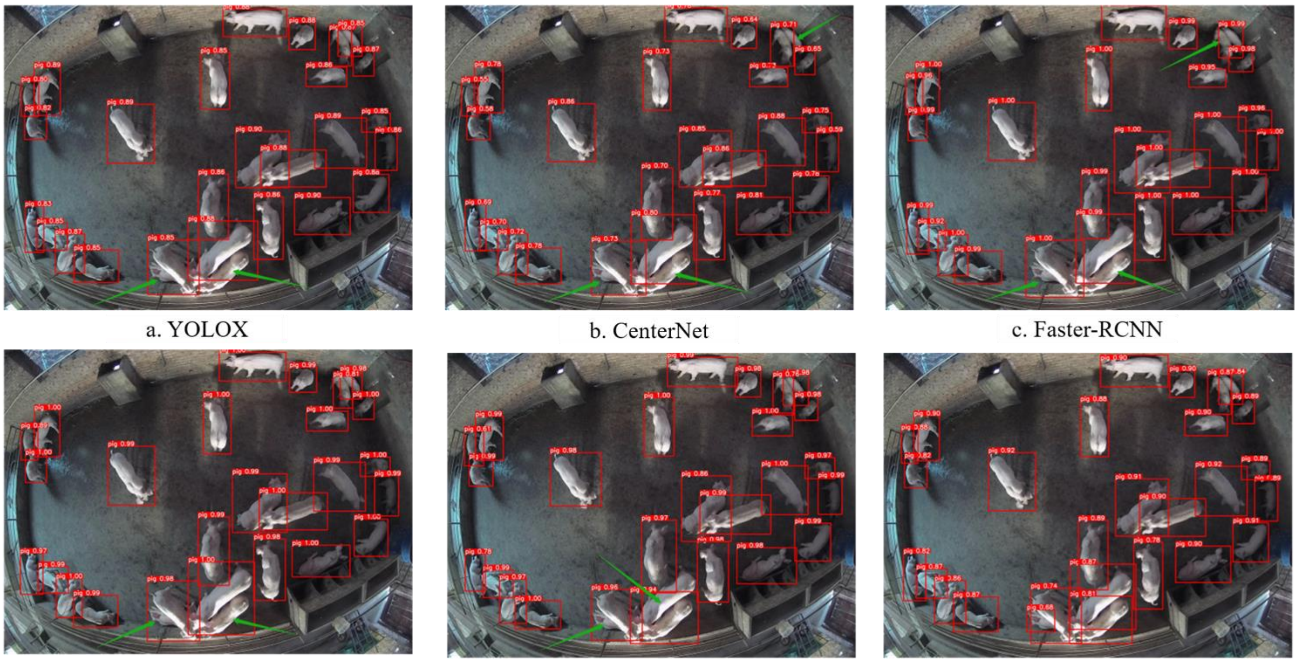
Fig. 3. Training accuracy curves of different models.

Table IV illustrates the detailed quantitative analysis of the model performance on the test dataset under same experimental configuration. It can be found that our proposed method outperforms other methods in terms of accuracy, speed, and computational efficiency. The mAP of our method achieved the highest value of 95.3%, while the FPS reached 154 frames/second. The computational complexity and number of parameters were also the lowest, with only 16.7 GFLOPs and 3.64M parameters, respectively.

To provide a more intuitive comparison of the performance of different models for individual pig detection in group feeding, Fig. 4 compares the detection results of the improved YOLOv5s model with those of other models. The green arrows in the figure indicate the missed pigs detected by each model. Compared to our model, the other models all showed missed detections, and the missed pigs in these models were all in heavily occluded, with two of them showing only a small part of their bodies, which is a small object detection problem. These results demonstrate that the improved YOLOv5s model performs better in detecting occluded and sticky pigs, as well as small objects, effectively improving the accuracy of individual pig detection in group feeding.

E. Ablation Study

To explore the effectiveness of the proposed improvements in this paper, we conduct extensive ablation study of C3S module, dual adaptive weighted PAN structure and small object detection layer. Table V shows that the C3S module significantly reduces the number of parameters, computation cost, and model layers while maintaining almost the same detection accuracy, demonstrating its effectiveness in improving the detection speed. Additionally, the proposed dual adaptive weighted PAN structure improves the precision, recall, and mAP of the model while keeping the computation cost, parameter count, and detection speed almost unchanged. By improving the detection layer, the model's recall and mAP are effectively increased with minimal reduction of detection speed. The improved YOLOv5s model shows a 3.8% increase in recall and a 2% increase in mAP, while reducing the parameter count by 48%, increasing FPS by 12.4%, and reducing model depth by 22%.



a. YOLOX b. CenterNet c. Faster-RCNN

d. YOLOv3 e. YOLOv4 f. Ours

Fig. 4. Comparison of the detection effect of different models.

TABLE IV. PERFORMANCE COMPARISON OF DIFFERENT OBJECT DETECTION MODELS

Model	Backbone	mAP (%)	Params (MB)	FLOPs(G)	FPS	Model size (MB)
Faster-RCNN	VGG16	86.28	136.7	369.9	20	521
YOLOv3	DarkNet53	92.19	61.5	65.52	47	235
YOLOv4	CSPDarkNet53	86.55	63.9	60	46	244
YOLOX	CSPDarkNet53	93.06	8.9	26.6	61	34
CenterNet	ResNet50	86.83	32.7	70	53	125
Ours	CSPDarkNet53	**95.3**	**3.64**	**16.7**	**154**	**8**

TABLE V. RESULTS OF ABLATION EXPERIMENT

Model	Precision (%)	Recall (%)	mAP (%)	Parameters (MB)	FLOPs (G)	FPS	Layers
YOLOv5s	94.8	85.3	93.3	7	15.8	137	235
YOLOv5s+a	94.7	84.8	93	4.4	**9.5**	**169**	**150**
YOLOv5s+a+b	**95.3**	87.5	94.1	4.57	9.9	164	168
YOLOv5s+a+b+c[b]	95.2	**89.1**	**95.3**	**3.64**	16.7	154	183

[b.] Note: a is the C3S module, b is the dual adaptive weighted PAN structure, and c is the small object detection layer.

F. Compared to Existing One-stage Pig Detection Methods

Table VI shows a comparison of the detection results achieved by our proposed improved YOLOv5s model on the test set, as well as the reported results of existing one-stage herd pig detection methods. Our dataset has a higher average number of pigs per image compared to the data reported in [10], [12] and [14]. Moreover, the pigs in our dataset exhibit higher levels of occlusion and adhesion, which increases the difficulty of object detection. Compared to the methods proposed in [10], [12] and [14], our proposed method achieved improvements of 9.45%, 1% and 0.97%, respectively, in terms of mAP.

TABLE VI. COMPARISON OF IMPROVED YOLOV5S AND EXISTING ONE-STAGE PIG DETECTION METHODS

Method	Average number of pigs per image	Model	mAP (%)
[10]	6	FPA-Tiny-YOLO	85.85
[12]	13	MF-CenterNet	94.30
[14]	9	YOLOv4	94.33
ours	31	Improved YOLOv5s	**95.30**

V. CONCLUSION

In this paper, an improved lightweight object detection method is proposed based on YOLOv5s, which has achieved higher accuracy and faster detection speed in high pig density scenarios with severe occlusion and adhesion. The lightweight C3S module proposed in this paper reconstructs the backbone and neck, resulting in a significant reduction in model parameters, computational complexity, and model depth. These modifications greatly enhance the detection speed to meet the requirements of real-time detection. The proposed dual adaptive PAN structure enhances the feature fusion capability of the neck, leading to improved detection accuracy. Furthermore, replacing the original large object detection layer with a small object detection layer in the detection stage significantly increases the recall rate and average detection precision.

Compared to existing methods for herd pig detection, our method simultaneously satisfies the requirements of high accuracy and real-time detection, making it deployable in practical group-raised farming scenarios and providing significant technical support for disease monitoring and pig counting. For future work, we expect to further enhance the feature extraction capabilities of the backbone network and streamline the model to construct a faster and more accurate object detection model for group-raised pig monitoring.

ACKNOWLEDGMENT

This work was supported by National Key R&D Program of China (Grant No. 2021YFD1300502).

REFERENCES

[1] National Bureau of Statistics of the People's Republic of China. China Statistical Yearbook [J]. Peking: China Statistics Press, 2022.

[2] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[3] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.

[4] Redmon J, Farhadi A. Yolov3: An incremental improvement [J]. arXiv preprint arXiv:1804.02767, 2018.

[5] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing, 2016: 21-37.

[6] Tian Z, Shen C, Chen H, et al. Fcos: Fully convolutional one-stage object detection[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 9627-9636.

[7] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[8] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.

[9] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks [J]. Advances in neural information processing systems, 2015, 28.

[10] Hongwen Y, Zhenyu L, Qingliang C. Multi-target detection based on feature pyramid attention and deep convolution network for pigs [J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2020, 36(11): 193-202.

[11] SHEN M, TAI M, CEDRIC O. Real-time detection method of newborn piglets based on deep convolution neural network [J/OL][J]. Transactions of the Chinese Society for Agricultural Machinery, 2019, 50(8): 270-279.

[12] Fang J, Hu Y, Dai B, et al. Detection of group-housed pigs based on improved CenterNet model [J]. Trans. Chin. Soc. Agric. Eng, 2021, 37: 136-144.

[13] Seo J, Ahn H, Kim D, et al. EmbeddedPigDet—Fast and accurate pig detection for embedded board implementations[J]. Applied Sciences, 2020, 10(8): 2878.

[14] Ahn H, Son S, Kim H, et al. EnsemblePigDet: Ensemble deep learning for accurate pig detection[J]. Applied Sciences, 2021, 11(12): 5577.

[15] Riekert M, Klein A, Adrion F, et al. Automatically detecting pig position and posture by 2D camera imaging and deep learning[J]. Computers and Electronics in Agriculture, 2020, 174: 105391.

[16] Li D, Zhang K, Li X, et al. Mounting behavior recognition for pigs based on Mask R-CNN[J]. Trans. Chin. Soc. Agric. Mach, 2019, 50: 261-266.

[17] Jocher G, Chaurasia A, Stoken A, et al. ultralytics/yolov5: v6. 1-tensorrt, tensorflow edge tpu and openvino export and inference[J]. Zenodo, 2022.

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

[19] Liu S, Qi L, Qin H, et al. Path aggregation network for instance segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 8759-8768.

[20] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.

[21] Zheng Z, Wang P, Ren D, et al. Enhancing geometric factors in model learning and inference for object detection and instance segmentation[J]. IEEE Transactions on Cybernetics, 2021, 52(8): 8574-8586.

[22] Yu J, Jiang Y, Wang Z, et al. Unitbox: An advanced object detection network[C]//Proceedings of the 24th ACM international conference on Multimedia. 2016: 516-520.

[23] Duan K, Bai S, Xie L, et al. Centernet: Keypoint triplets for object detection[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 6569-6578.

[24] Ge Z, Liu S, Wang F, et al. Yolox: Exceeding yolo series in 2021[J]. arXiv preprint arXiv:2107.08430, 2021.