2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Novel Defect Detection for a Badminton Shuttlecock Based on Improved YOLOv8 with RepVGGBlock

Yujie Li, Xin Li, Qiuyuan Gan, and Benying Tan(✉)

School of Artificial Intelligence, Guilin University of Electronic Technology, China
by-tan@guet.edu.cn

Abstract. In badminton shuttlecock secondary recycling, manual selection is easily affected by human factors, such as fatigue and attention lapses. This leads to low efficiency and makes it impossible to satisfy large-scale refurbishment requirements. This paper proposes YOLOv8n_RepVGG, which is an enhanced badminton shuttlecock classification method that uses a modified YOLOv8n architecture. We used RepVGGBlock modules in the backbone network to improve the representational capacity of the model. Compared with the baseline YOLOv8, the proposed model achieved a precision of 91.1% with an increase of 3.1% and mean average precision (mAP50) of 87.2% with an increase of 1.2%. The proposed approach not only propels technological advances in badminton shuttlecock reconditioning processes but also contributes significantly to global sustainability initiatives through enhanced resource optimization.

Keywords: Defect detection for badminton shuttlecock, YOLO, RepVGGBlock, Data augmentation, Deep learning.

1 INTRODUCTION

Badminton, a popular indoor sport, primarily uses goose and duck feathers in badminton shuttlecocks. Goose feather badminton shuttlecocks with stable flight, superior hardness, and impact resistance are mainly used in professional tournaments. Duck feathers, which are heavier, have curved rachises and are arranged in dense vanes that loosen over time, are commonly used in club training and recreational matches. The global rise in badminton has spurred demand for badminton shuttlecocks, driving the growth of the reuse and refurbishment industries. Heightened environmental awareness and sustainable development efforts have drawn attention to the recycling of badminton shuttlecocks. Secondary use extends the useful lives of products, reduces resource waste, and facilitates ecological conservation. However, traditional manual selection is inefficient and inaccurate for large-scale refurbishments, as human operators are prone to fatigue-induced errors and attention deficits, which cause workforce inefficiencies and heavy environmental burdens.

Traditional object detection algorithms have several technical limitations in badminton shuttlecock recognition and classification. The two-stage detection framework of the Faster R-CNN [1] achieves high-precision object localization using its region

proposal network (RPN). However, its multistage network architecture significantly increases its computational complexity. Therefore, it is unsuitable for use in real-time detection in industrial applications.

Single-stage detectors, represented by YOLO [2] and proposed by Redmon et al., exhibit superior real-time performance. However, they produce significant false negatives for sub-3 mm surface defects. Some examples include broken barbs and creases. These false negatives occur because of insufficient shallow feature extraction, which degrades the small-target representation. The transformer-based Detection Transformer (DETR) framework [3] introduced by Carion et al. achieved global feature modeling through self-attention mechanisms. However, the end-to-end training paradigm depends heavily on large-scale annotated datasets. This is a critical constraint, given the scarcity of defective badminton shuttlecock samples, particularly for rare defect categories. This issue severely compromises the generalization capabilities of the model.

Lightweight variants, such as YOLO-HGNet [4] with integrated depthwise separable convolutions, enhance inference speed via parameter reduction. Moreover, their underdeveloped multiscale feature fusion mechanisms result in unstable detection performance for hierarchical feather defects. Examples include concurrent root fractures and tip abrasion.

To address these challenges, this study focused on developing defect-detection methods for badminton shuttlecocks. YOLOv8 is a state-of-the-art object detection algorithm that offers exceptional real-time performance with high accuracy for complex visual-recognition tasks. We proposed an improved YOLOv8 with RepVGGBlock that can precisely identify critical badminton shuttlecock characteristics. The characteristics of badminton shuttlecocks include the verification of new conditions, feather loss (lose_feather), and fracture point (break) detection. Fig. 1 presents a flowchart of the proposed method, which includes key stages, such as data augmentation, model training, and testing. This automated approach not only enhances operational efficiency but also significantly reduces the labor costs and resource waste associated with manual inspection processes.

This study not only drives technological innovation in badminton shuttlecock refurbishment but also offers substantial support for societal green transformation, marking a strategic shift toward intelligent and sustainable industry practices. Through the adoption of advanced technological solutions, we demonstrated effective strategies for addressing resource scarcity and environmental challenges, thus making concrete contributions to achieving the development goals. The key contributions of this study are as follows.

- We propose a novel defect detection YOLOv8n_RepVGG for badminton shuttlecocks that can quickly identify badminton shuttlecock features in various conditions.
- The proposed YOLOv8n_RepVGG employs RepVGGBlock in the backbone network to improve the representational capacity of the model and increase the detection performance.
- Data augmentation is used to increase the diversity and complexity of the dataset, which can help the proposed model learn robust features.

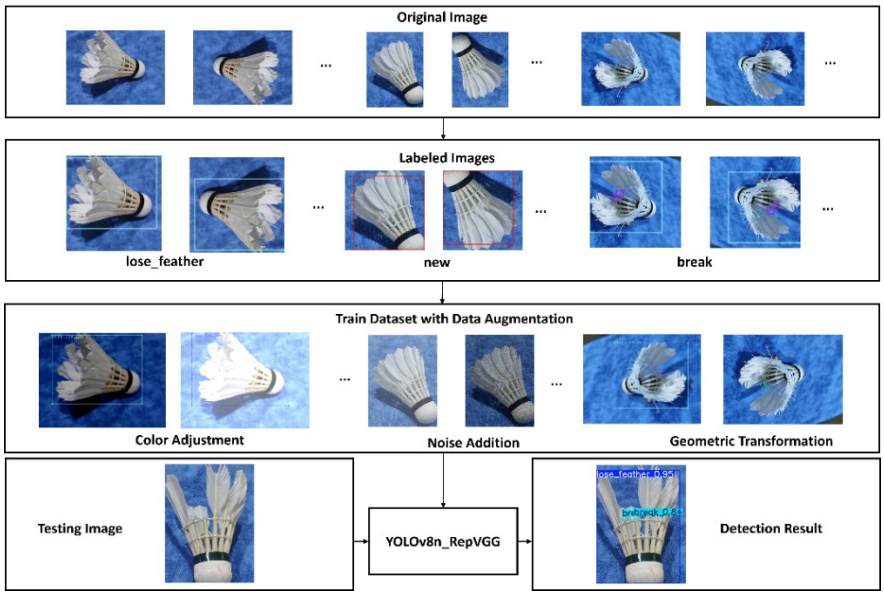

Fig. 1. Flowchart of the proposed defect detection for badminton shuttlecock.

2 RELATED WORK

In recent years, deep learning has significantly advanced object detection and produced several representative algorithms. According to their detection mechanisms and features, these algorithms can be classified as single-stage, two-stage, anchor-based, or anchor-free. Two-stage detection algorithms, represented by Mask R-CNN [5] and Faster R-CNN [6], utilize an RPN to generate the candidate boxes. They refined the boxes using precise classification and regression. These algorithms provide high localization accuracy. For example, Faster R-CNN attains a mAP of 82.3%, but its inference speed is low at approximately 8 FPS. Thus, it is suitable for use in scenarios in which the real-time performance is not crucial.

Single-stage detection algorithms, such as the YOLO [7] series and SSD [8], adopt an end-to-end regression mechanism to predict bounding boxes and class probabilities directly. For instance, YOLOv8 can achieve a real-time detection speed of 45 FPS with merely 2.7M parameters (in the YOLOv8n version), excelling in lightweight and efficient performance. In the small-object detection of unmanned aerial vehicle images, Li et al. [9] improved YOLOv7 to enhance its accuracy and efficiency. Li et al. [10] proposed an intelligent retrievable object tracking system that significantly improved the ability to locate lost objects accurately and efficiently in indoor home environments. Wang et al. [11] extensively studied and optimized YOLOv7, achieving a new speed--accuracy trade-off for real-time object detection. These studies provide crucial references for future upgrades to the YOLO series.

In addition to classic algorithms, such as Mask R-CNN, Faster R-CNN, SSD, FCOS, and the YOLO series, other methods have also significantly contributed to object

detection. For example, RetinaNet [12] introduces focal loss to address class imbalances and improve the performance of single-stage detectors. Deformable DETR [13] combines deformable convolutions with transformer mechanisms to achieve excellent detection in complex scenarios. EfficientDet [14] optimizes the balance between speed and accuracy using a neural architecture search. These methods provide innovative ideas for different aspects of object detection and lay the groundwork for further performance improvement.

In this study, we selected the YOLOv8 model [15] for an efficient large-scale badminton shuttlecock classification. This combines the advantages of the YOLO series to enable fast and accurate badminton shuttlecock classification. YOLO is an end-to-end object detection system that makes predictions based on global image information by dividing the image into grids and predicting the bounding boxes and classes for each grid cell. The official code of YOLOv8 offers five versions (i.e., n, s, m, l, and x) with increasing network widths and depths, resulting in improved detection speeds and accuracies. To achieve high-precision lightweight detection, we selected the YOLOv8n model as the baseline and made several improvements.

3 METHOD

This section discusses defect detection in the badminton shuttlecock by emphasizing model upgrades that enhance the identification of broken feather points. These upgrades were achieved by integrating RepVGGBlock modules into the backbone network to replace the original downsampling components, strengthen the feature extraction of the model, and significantly improve the damaged feather recognition precision. Consequently, the model is more robust in complex environments and provides a more efficient and accurate solution for badminton shuttlecock recycling and reuse.

3.1 YOLOv8n_RepVGG

To enhance the badminton shuttlecock detection of YOLOv8, this study introduced an improved YOLOv8n_RepVGG model, as depicted in Fig. 2. By integrating RepVGG modifications, this model boosts object classification accuracy.

The model comprises four parts: input, backbone, neck, and head. The input end uses adaptive image scaling and mosaic data augmentation to bolster model robustness. The backbone comprised Conv, RepVGGBlock, C2f, and SPPF modules. The Conv module uses 3×3 kernels for local feature extraction and 1×1 kernels for channel adjustment and feature fusion. Batch normalization accelerates training and enhances stability, whereas the SiLU activation function ensures a smoother gradient flow and faster convergence. Unlike the C3 module of YOLOv5, the C2f module enriches the gradient flow with skip connections and additional split operations. Specifically, the input feature map is split into two parts: one remains unchanged, and the other is processed through multiple bottleneck modules with 3×3 depthwise separable convolutions. Finally, Concat fuses the two parts to preserve the rich gradient information, helping the model learn more discriminative feature representations. The SPPF module employs

pyramid pooling at different scales (i.e., 5×5, 9×9, 13×13) to extract features and thus enable the network to capture multiscale context information, which is crucial for the object recognition and localization of varying object sizes in object detection tasks.

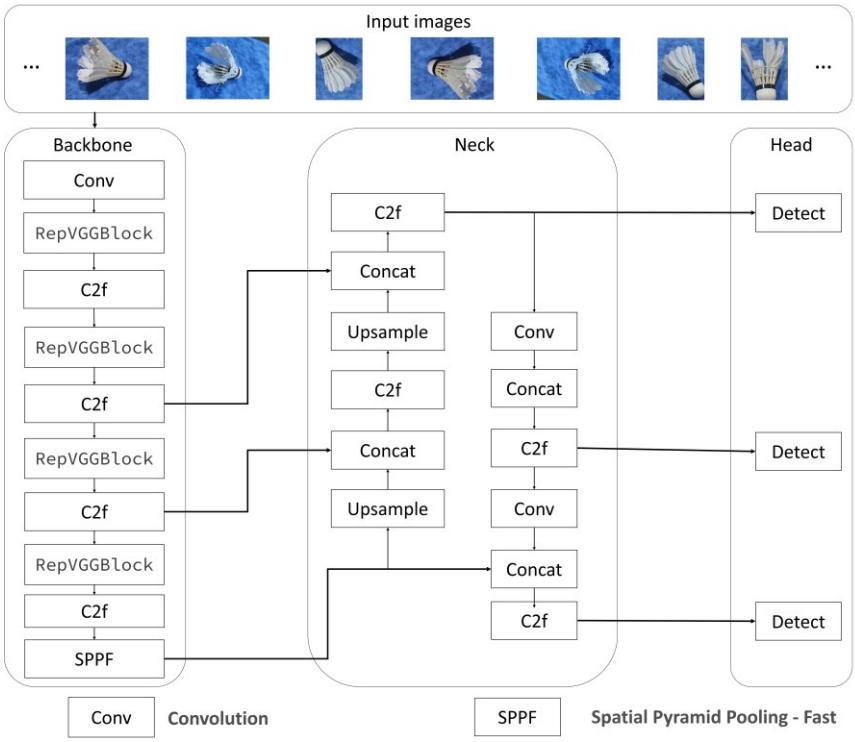

Fig. 2. The structure diagram of the proposed YOLOv8n_RepVGG.

The neck uses an improved path aggregation network (PAN) architecture for defect detection in a badminton shuttlecock, integrating multiscale features via bidirectional fusion to balance the small-object detection accuracy and inference efficiency. The key components of this improved PAN architecture are C2f modules, high-compression Conv layers, and Upsample modules. The C2f modules have fewer bottlenecks for gradient optimization. High-compression Conv layers reduce the channel dimensions for efficiency. The Upsample modules used interpolation for feature alignment. This design improves the robustness of complex scenes through a dynamic multiscale semantic fusion.

In the defect detection for a badminton shuttlecock, the detection head uses a decoupled head architecture. This architecture separates the classification and bounding-box regression tasks to resolve parameter conflicts as well as enhance feature expression and training efficiency. The detection head employs dynamic channel adjustment for multiscale inputs and combines the spatial and channel attention mechanisms. The regression branch introduces distribution focal loss (DFL) for the discrete distribution modeling of bounding box predictions, which improves the localization accuracy. In

addition, YOLOv8 adopts an anchor-free mechanism that involves directly predicting the distances from the object centers to the boundaries. These advantages include simplified post processing and reduced hyperparameter dependence.

3.2 RepVGGBlock

For the RepVGGBlock module in the backbone, we selected a simple yet powerful CNN architecture. This architecture has a VGG-like inference time body composed solely of stacked 3×3 convolutions and ReLUs. In contrast, the training-time model features a multi-branch topology. This decoupling of the training and inference architectures was achieved through structural reparameterization. Hence, it was named RepVGG. RepVGG is a classification network based on VGG with several key improvements.

- We introduce two residual structures into the Block blocks of the VGG network: the identity branch and 1×1 branch. These residual structures' multiple branches provide extra gradient flow paths during training, similar to training and fusing several networks into one.
- During the inference phase, we employ an operation fusion strategy. This strategy converts all network layers to Conv 3×3. As a result, we obtain a simplified RepVGG network architecture. In this architecture, the entire network consists of Conv 3×3 + ReLU stacks. This facilitates efficient model inference and acceleration.

A partial expression of the RepVGG network architecture during training and inference was presented in [16].

3.3 Data augmentation

We used data augmentation to improve the generalization and robustness of the model using three methods: color adjustment, noise addition, and geometric transformation. The color adjustments included rb, rs, and rc. Noise addition included gn, pn, and sn. Geometric transformations involved cc, fh, and fv.

- Center Crop (cc): Crops the image center to a square and adjusts the corresponding bounding boxes.
- Flip Horizontal (fh): Randomly flips images horizontally and adjusts bounding box coordinates.
- Flip Vertical (fv): Randomly flips images vertically and adjusts bounding box coordinates.
- Add Gaussian Noise (gn): Adds Gaussian noise to images to enhance data diversity.
- Add Pepper Noise (pn): Adds pepper noise to increase visual sample complexity.
- Add Salt Noise (sn): Adds salt noise to increase image randomness.
- Random Brightness (rb): Randomly adjusts image brightness.
- Random Contrast (rc): Randomly adjusts image contrast.
- Random Saturation (rs): Randomly adjusts image saturation.

4 EXPERIMENTS

In comparative experiments, we tested five representative and high-performing models: YOLOv5n, YOLOv8n, YOLOv10n, YOLOv11n, and YOLOv12n. Using the same dataset and evaluation metrics, we assessed the Precision, Recall, Giga Floating-point Operations Per Second (GFLOPs), mAP50, and mAP50-95 to compare their differences.

4.1 Dataset

The dataset for this project consists of three categories of labels: lose_feather, break, and new, to simulate the inspection and quality control processes of badminton shuttlecocks in enterprises. Initially, 1024 badminton shuttlecock images were collected. To enhance the quality and diversity of the dataset, we applied data augmentation techniques to expand the dataset to 2683 images. Following an 8:2 random split, the training set contained 2146 images, and the test set contained 537 images. Table 1 details the label distribution, showing the number of labels for each class after processing as well as their allocation in the training and validation sets.

Table 1. Label distribution across data subsets.

Class	Processed	Train	Validation
lose_feather	1481	1170	311
break	1045	830	215
new	1203	977	226

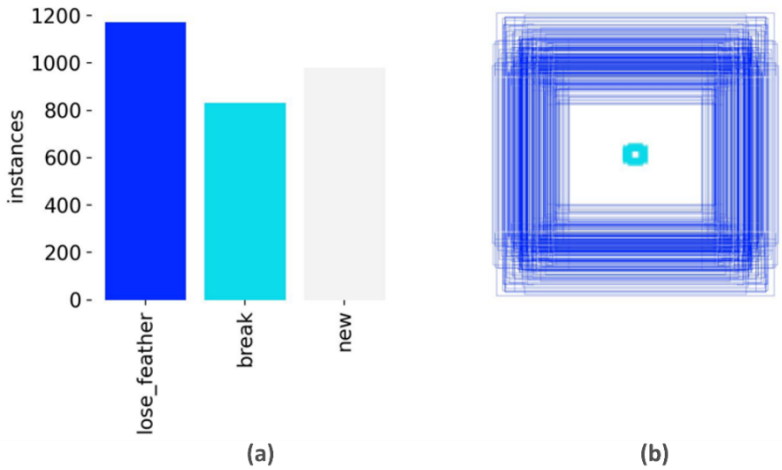

Fig. 3. (a): training set data volume distribution. (b): bounding box size-quantity relationship in training set.

Fig. 3 (a) depicts the data volume distribution for each category in the training set, which indicates the number of samples per category. Fig. 3 (b) illustrates the relationship between the bounding box size and quantity in the training set, displaying how the box numbers change with size.

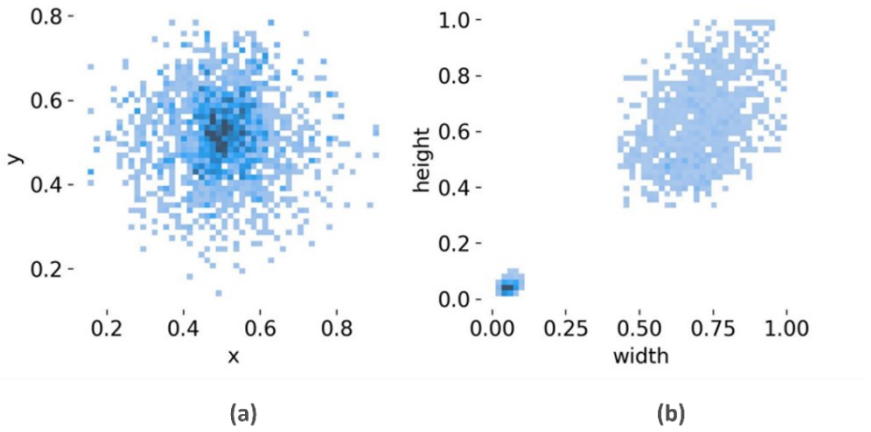

Fig. 4. (a): distribution of bounding box center points in images. (b): aspect ratio distribution of objects in images.

Fig. 4 (a) presents the distribution of the bounding box center points relative to the entire image, showing their frequencies across different regions. Fig. 4 (b) shows the aspect ratio distribution of objects relative to the entire image, reflecting variations in height-to-width proportions across categories and instances. All badminton shuttlecock images came from actual badminton shuttlecocks used in courts, ensuring that the dataset reflected real-world conditions. This design improves model training and enhances the accuracy and reliability of the detection system in practical applications to provide a robust data foundation for automatic badminton shuttlecock detection and repair.

Data augmentation was performed to generate additional training images. As shown in Fig. 5, the first image is the original image, and the following nine images show the results after data enhancement.

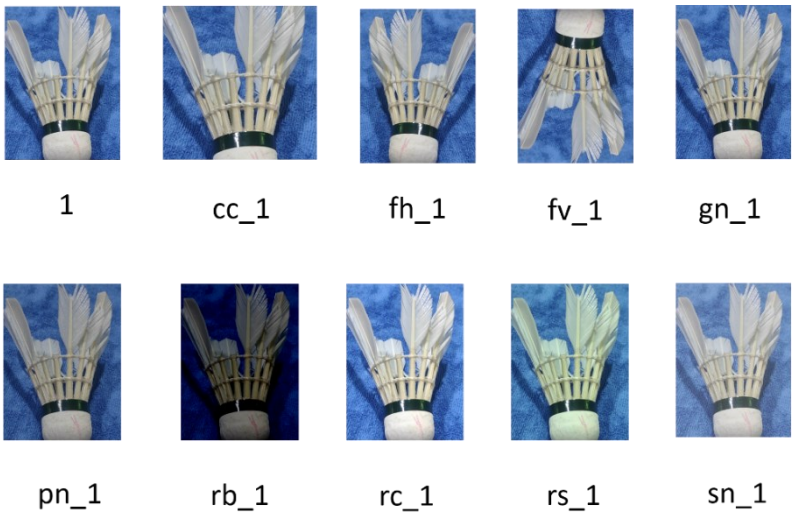

Fig. 5. Comparison of example images before and after data augmentation.

4.2 Evaluation Metric

To validate the effectiveness of our model experimentally, we set the evaluation metrics during training as Precision, Recall, GFLOPs, mAP50, and mAP50-95. The specific formulae are as follows:

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

$$mAP = \frac{1}{n}\sum_{i=0}^{n} P_{AP}(i) \tag{3}$$

where TP represents the number of true positives predicted by the model, FP represents the number of false positives predicted by the model, FN represents the number of false negatives predicted by the model, Precision is the proportion of actual positive cases among those predicted as positive, Recall is the proportion of cases predicted as positive among the actual positive cases, GFLOPs is a computational complexity metric, indicating the number of billions of floating-point operations executed per second, used to evaluate the computational demands and performance of models or hardware. mAP50 represents the average precision of each category when the confidence threshold is 0.5, and mAP50-95 represents the average precision after weighting the average of the confidence thresholds from 0.5 to 0.95 with a step length of 0.05. According to the definition of an AP:

$$AP = \int_0^1 P(R)dR \tag{4}$$

The Average Precision (AP) is the area under the precision-recall curve, with recall on the x-axis and precision on the y-axis. All experiments were conducted on a local computer with consistent hyperparameter settings (batch sizes of 16 and 200 epochs). To ensure validity, all baseline models were trained from scratch without pretraining. The experimental environment is detailed in Table 2.

Table 2. Hardware and software configuration.

Configuration Environment	Version Environment
Operating System	Window 11
Machine Learning Framework	PyTorch 2.4.1
GPU Acceleration Library	cuda11.8
Programming Environment	Python 3.8.19
GPU	NVIDIA GeForce RTX 3050

4.3 Experimental results

Tables 3-6 present the experimental results for YOLOv8n_RepVGG and the other baseline models with different labels.

Table 3. Comparison of overall experimental results among different models.

Model	Precision	Recall	mAP50	mAP50-95	GFLOPs
YOLOv5n	**93.6%**	79%	84.2%	71%	7.1
YOLOv8n	88%	**84.8%**	86%	72.6%	8.1
YOLOv10n	82.3%	82.4%	82.8%	69.6%	8.2
YOLOv11n	91.1%	81.2%	85.4%	71.6%	6.3
YOLOv12n	90.3%	80.3%	83.9%	68.1%	5.8
Ours	91.1%	84.2%	**87.2%**	**72.9%**	8.2

Table 3 lists the overall performance of different models. Among the recent YOLO-based methods, the proposed YOLOv8n_RepVGG achieved the highest mAP50 and mAP50-95 among the recent YOLO-based methods. Our model has a GFLOPs of 8.2, similar to YOLOv8n's 8.1. This indicates that our method achieves performance improvement while essentially maintaining the same level of computational complexity. Compared with the most recent YOLOv12n, the proposed method can obtain an increase of 3.3% (83.9%-87.2%) in mAP50 and 4.8% (68.1%-72.9%) in mAP50-95.

Table 4 compares the results of the different models for the lose_feather label. The proposed YOLOv8n_RepVGG achieved a higher mAP50 than did the other YOLO-based methods. Compared with the most recent YOLOv12n, the proposed method can obtain a slight increase of 0.5% (98.5%-99%) in mAP50 and 2.8% (86.4%-89.2%) in mAP50-95.

Table 4. Comparison of experimental results for the lose_feather label among different models.

Lose_feather	Precision	Recall	mAP50	mAP50-95
YOLOv5n	97.1%	99.4%	98.9%	89.1%
YOLOv8n	96.8%	98.4%	98.8%	**90.1%**
YOLOv10n	93.9%	99.4%	**99%**	89.2%
YOLOv11n	**97.3%**	98.4%	98.9%	89.2%
YOLOv12n	95.4%	**99.7%**	98.5%	86.4%
Ours	97.2%	99%	**99%**	89.2%

Table 5. Comparison of experimental results for the break label among different models.

Break	Precision	Recall	mAP50	mAP50-95
YOLOv5n	**85.6%**	41.3%	54.7%	32.8%
YOLOv8n	72.4%	**58.1%**	60.3%	35.1%
YOLOv10n	57%	50.5%	50.7%	27.9%
YOLOv11n	79.8%	47.7%	58.4%	33.9%
YOLOv12n	79.8%	44.7%	54.4%	28.7%
Ours	79.2%	56.7%	**63.5%**	**36.8%**

Table 6. Comparison of experimental results for the new label among different models.

New	Precision	Recall	mAP50	mAP50-95
YOLOv5n	**98.2%**	96.2%	99.1%	91.2%
YOLOv8n	94.8%	**97.8%**	99%	92.6%
YOLOv10n	96.2%	97.3%	98.8%	91.6%
YOLOv11n	96.2%	97.3%	98.8%	91.5%
YOLOv12n	95.6%	96.6%	98.9%	89.3%
Ours	96.9%	96.8%	**99.2%**	**92.6%**

Table 5 compares the results of the various models for the break label. The proposed model significantly improved mAP50 and mAP50-95, outperforming all the comparison models. Compared with the most recent YOLOv12n, the proposed method obtained a significant increase of 9.1% (54.4%-63.5%) in mAP50 and 8.1% (28.7%-36.8%) in mAP50-95.

Table 6 presents the results of the different models on the new label. The proposed model also achieved the highest mAP50 and mAP50-95 values, outperforming all the comparison models. Compared with the most recent YOLOv12n, the proposed method obtained an increase of 0.3% (98.9%-99.2%) in mAP50 and 3.3% (89.3%-92.6%) in mAP50-95.

Moreover, the intuitive results are presented in Fig. 6. As shown in Fig. 6, the first row shows the results obtained by the baseline YOLOv8n, whereas the second row shows the results obtained by YOLOv8n_RepVGG. The results demonstrate that the

proposed model identifies critical break defects (e.g., column 1) that are not detected by the baseline. The proposed model can achieve higher prediction confidence in most categories (lose_feather, break, new) than YOLOv8n.

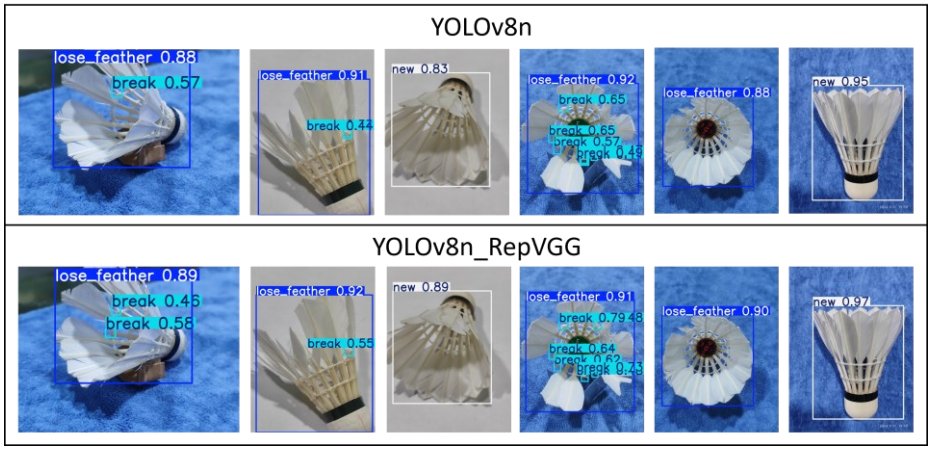

Fig. 6. Comparison results between YOLOv8n and the proposed YOLOv8n_RepVGG.

4.4 Discussion

The YOLOv8n_RepVGG model proposed in this study has made significant progress in the detection and recognition of badminton shuttlecocks. We conducted an in-depth analysis of the advantages and disadvantages of the model.

- The YOLOv8n_RepVGG model excels in detecting specific types of badminton shuttlecock defects. Notably, it achieves a remarkable mAP50 of 63.5% for the break label, significantly outperforming other models. This enhanced capability is crucial for quality control in badminton shuttlecock refurbishment, as it enables more accurate identification of broken feather points.
- The use of data augmentation techniques, such as cropping, flipping, and brightness adjustment, significantly boosts the robustness and generalizability of the model. These techniques increase the diversity of the dataset and enable the model to maintain high detection accuracy across various lighting conditions, angles, and backgrounds, ensuring reliable performance in real-world applications.
- Limitation: The YOLOv8n_RepVGG model heavily relies on the quantity and quality of the training data. Limited or insufficiently diverse datasets can restrict its performance. In scenarios in which data collection is challenging or the available data are limited, the model may not achieve optimal detection accuracy, potentially resulting in misclassifications.

5 CONCLUSION

In this study, we proposed a YOLOv8n_RepVGG model for badminton shuttlecock detection. Through feature extraction using the employed RepVGGBlock, the model can effectively identify and classify badminton shuttlecocks of different qualities. Moreover, we used data augmentation, including color adjustment, noise addition, and geometric transformation, to significantly enhance the generalization and robustness of the proposed model. The experimental results showed that data augmentation significantly improved the detection accuracy and robustness of the model. Future research will focus on expanding the dataset by enhancing data diversity and optimizing the model architecture to meet potential challenges and improve the detection performance. This research demonstrates the effectiveness of YOLOv8 in badminton shuttlecock detection and lays a solid foundation for relevant applications.

References

1. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28 (2015)
2. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
4. Yang, W., Jiang, M., Fang, X., Shi, X., Guo, Y., Al-qaness, M.A.: A high-precision and efficient method for badminton action detection in sports using you only look once with hourglass network. Engineering Applications of Artificial Intelligence 137, 109177 (2024)
5. Fizaine, F.C., Bard, P., Paindavoine, M., Robin, C., Bouyé, E., Lefèvre, R., Vinter, A.: Historical text line segmentation using deep learning algorithms: Mask-rcnn against u-net networks. Journal of Imaging 10(3), 65 (2024)
6. Hou, J., Che, Y., Fang, Y., Bai, H., Sun, L.: Early bruise detection in apple based on an improved faster rcnn model. Horticulturae 10(1), 100 (2024)
7. Cui, Y., Guo, D., Yuan, H., Gu, H., Tang, H.: Enhanced yolo network for improving the efficiency of traffic sign detection. Applied Sciences 14(2), 555 (2024)
8. Hou, H., Guo, M., Wang, W., Liu, K., Luo, Z.: Improved lightweight head detection based on ghostnet-ssd. Neural Processing Letters 56(2), 126 (2024)
9. Li, Y., Wang, Y., Ma, Z., Wang, X., Tang, Y.: Sod-uav: small object detection for unmanned aerial vehicle images via improved yolov7. In: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 7610–7614. IEEE (2024)
10. Li, Y., Wang, Y., Ma, Z., Wang, X., Tan, B., Ding, S.: An intelligent retrievable object-tracking system with real-time edge inference capability. IET Image Processing 19(1), e13297 (2025)
11. 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: Proceedings of the

IEEE/CVF conference on computer vision and pattern recognition. pp. 7464–7475 (2023)
12. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
13. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 (2020)
14. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10781–10790 (2020)
15. Yang, T., Zhou, S., Xu, A., Ye, J., Yin, J.: An approach for plant leaf image segmentation based on yolov8 and the improved deeplabv3+. Plants 12(19), 3438 (2023)
16. Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., Sun, J.: Repvgg: Making vgg-style convnets great again. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13733–13742 (2021)