Proceedings of 2023 IEEE
International Conference on Mechatronics and Automation
August 6 - 9, Harbin, Heilongjiang, China

Gear Fault Detection Method Based on the Improved YOLOv5

Xin Wan, Manyi Wang
School of Mechanical Engineering
NanJing University of Science and Technology
XiaoLingWei Street, Nanjing, China
Email:2407967161@qq.com, manyi.wang@njust.edu.cn

2023 IEEE International Conference on Mechatronics and Automation (ICMA) | 979-8-3503-2084-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICMA57826.2023.10215657

***Abstract* – Gears are used as transmission elements in a wide range of industries, so detecting faults in them is important. Current deep learning-based fault detection is difficult to apply to industrial embedded devices due to the complexity of the model and the huge computational effort. To address this problem, we propose a lightweight gear fault detection model, LG-YOLOv5. To obtain a lightweight network, the introduction of ShuffleNetV2 and GSConv. Then, to ensure excellent detection performance, we integrate a multi-span hybrid spatial pyramid pooling model, attention mechanism modules and cross-scale feature pyramids to improve the detection performance. Finally, to evaluate the gear fault detection capability of the LG-YOLOv5 on the Rockchip RK3568 embedded platform. Image acquisition to create a gear fault dataset. Experimental results show that the LG-YOLOv5 model has a volume of 8.8M, which is only 61.5% of the YOLOv5 model, a computational cost of 13.6% of the YOLOv5, a 45% increase in detection speed and a 1.5% increase in accuracy, and is able to accurately identify gear faults such as wear, bulging and missing tooth.**

Index Terms - Gear fault detection, lightweight detector, YOLOv5, object detection.

I. INTRODUCTION

Gears are important components in mechanical systems for transmitting power and changing direction, and are widely used in industry, defence, aerospace and other fields. Due to the harsh working environment and intensity of work, gears are prone to wear, missing tooth and other faults, which directly affect the normal operation of the overall system. Therefore, it is important to carry out gear fault detection in real time and accurately.

The traditional fault detection method is to collect the sensor signal and then filter, feature extraction and classification to achieve detection [1]. Xiao HL et al. utilized the global search capability of genetic algorithms and the optimal parameter method of support vector machines to achieve fault detection [2]. Mariela et al. extracted the features of gears from vibration signals using fast Fourier transform and wavelet packet decomposition methods and used genetic algorithms to achieve fault detection [3]. M. Zhao et al. designed a dynamic weighted wavelet coefficient to achieve planetary gearbox fault detection [4]. While these methods are effective in identifying faults, the data collection and processing process is cumbersome and poor in real time.

Fault detection based on target detection allows gear faults to be discerned from image information. Target detection networks can be divided into two-stage and one-stage methods according to their structure. The two-stage approach has high detection accuracy but is slow. The one-stage approach uses end-to-end target detection and has a faster detection speed to meet real-time requirements. Liya Yu et al. proposed a gear defect online detection model S-YOLO for image acquisition in complex backgrounds [5]. Dejun Xi et al. constructed a two-stage network using YOLOv5 and an improved Deeplabv3+ to achieve real-time detection of raised faults [6]. Z. Zheng et al. proposed BNA-Net to improve YOLO and achieve defect detection with high accuracy [7].

In industrial application scenarios, only embedded platforms with limited memory and computing power can be used, and the above methods are computationally intensive. In order to achieve accurate and fast detection, this paper proposes a lightweight fault detection network, LG-YOLOv5.

The main contributions of this paper are as follows:

1) To lighten the network model, we replaced the YOLOv5 Backbone with a ShuffleNetV2 lightweight network [8] and used GSConv to replace Conv in Neck [9], thereby significantly reducing the computational effort.

2) To ensure the detection performance of the model, we propose an improved BiFPN [10]. The direct link between the input feature layer and the output feature layer is enhanced to achieve more feature information retention.

3) The extraction of gear fault features is enhanced using Shuffle Attention [11] and NAM [12]. We also propose a multi-span hybrid spatial pyramidal pooling model, LR-SPPF, to increase the contextual information of the sensory field.

To confirm the validity of the above contributions, we produced a gear failure dataset and conducted an extensive ablation study [13]. In addition, we have ported LG-YOLOv5 to an embedded platform to verify the performance of gear fault detection in a real-world environment.

II. NETWORK STRUCTURE OF YOLOV5

YOLO is an algorithm proposed by Joseph Redmon for target detection [14]. After years of development, YOLO has iterated to YOLOv5 [15], which has significantly improved detection accuracy and speed after incorporating the advantages of various networks. It is also a significant reduction in model size compared to its predecessor YOLO. Suitable for applications in all types of target detection

tasks.The YOLOv5 network structure is shown in Fig. 1 and consists of Input, Backbone, Neck and Output.

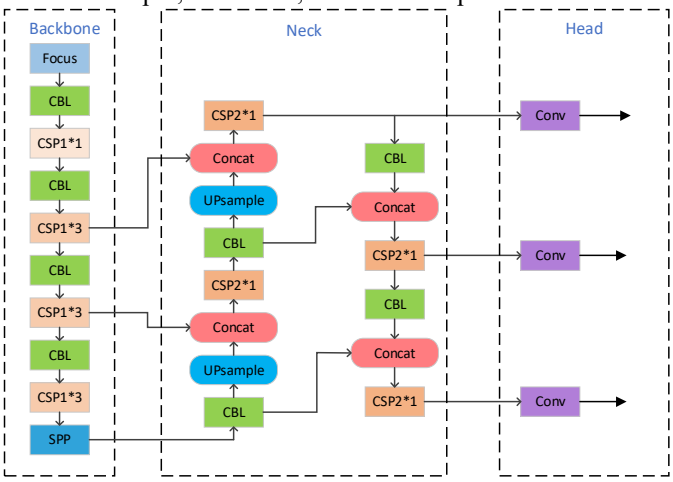

Fig. 1.YOLOv5 network architecture diagram.

On the input side of YOLOv5, images are pre-processed using Mosaic data enhancement and adaptive image scaling [16]. Mosaic data enhancement increases small target samples, expands the dataset and improves network robustness and training speed by randomly scaling, cropping and lining up multiple images to stitch them together into a single image. Adaptive image scaling minimises the black edges of image fill edges and reduces information redundancy.

Backbone consists of a Focus structure and a CSP structure [17]. Focus slices and stitches the input image and convolves it to obtain a bipartite down sampling feature map with no information loss. CSP uses multiple residual structures to increase the gradient value of backpropagation between layers and avoid gradient disappearance.

The Neck part uses a structure of FPN+PAN. FPN uses a top-down lateral connection to build out a high-level semantic feature graph at all scales. A bottom-up feature pyramid is later added, which includes two PAN structures that convey strongly localized features.

The output section uses CIOU for bounding box prediction [18]. CIOU calculates the difference between the predicted and real boxes by multi-dimensionality and works better than the previous generation of IOU after adding the aspect ratio calculation.

III. Improvement of YOLOv5 Network

The main problem with deep learning target detection models is the large computational volume [19], which can face a lack of memory with limited hardware platform resources, resulting in long response times that cannot meet the requirements of low-latency real-time detection. The LG-YOLOv5 proposed in this paper can effectively solve this problem, the network structure is shown in Fig. 2.

A. Backbone Improvement

Current research on model light weighting falls into two main directions: direct compression of complex models [20], and redesign of small models [21]. Either study aims to reduce model size and improve model speed while maintaining model performance.

In order to improve the speed of YOLOv5 on embedded platforms, we introduced ShuffleNetV2 to improve Backbone,

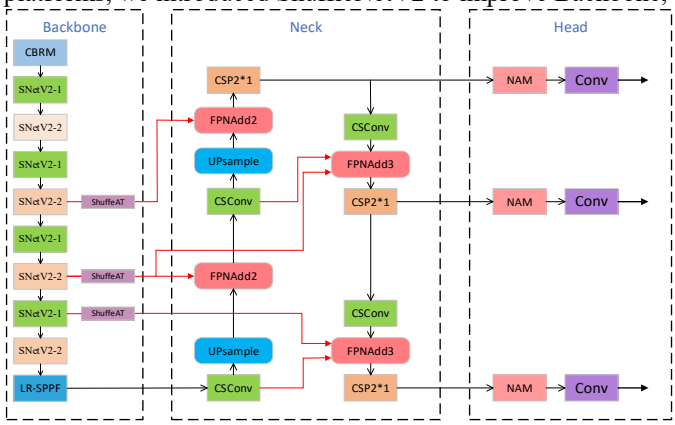

Fig. 2.LG-YOLOv5 network architecture diagram.

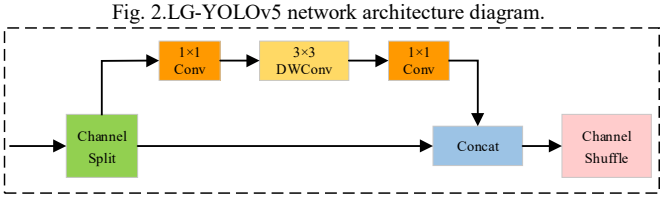

Fig. 3.Base unit network structure.

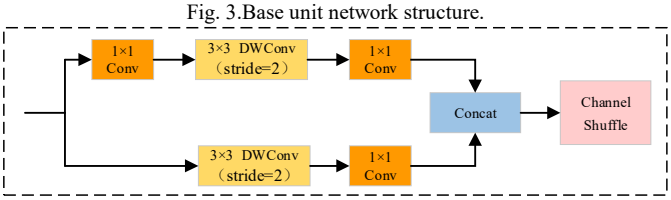

Fig. 4.Down sampling unit network structure.

taking into account the cost of memory access and the characteristics of embedded platforms.

ShuffleNetV2 mainly consists of the base unit Fig. 3 and the down sampling module Fig. 4. The base unit is the basic building block of the network structure. Through the Channel Split operation, the number of channels of the input feature map is divided into two equally, and after convolution operation is performed on one part, the two parts are concat, so as to achieve equal input and output feature matrix channels and minimize MAC. Finally, Channel shuffle is used to rearrange the channels and to exchange information between groups.

The down sampling module removes the channel Split and sets the DWConv step size to 2. This operation halves the feature map size and doubles the number of channels, enabling efficient information transfer and feature extraction. By combining the two modules, the network structure can maintain good performance while significantly reducing the number of parameters and computational cost.

B. SPP Improvement

In YOLOv5, Glenn Jocher improved the SPP module proposed by K. He [22] into the SPPF module. By changing the parallel computation to serial computation, the amount of repeated computation is greatly reduced and the computation speed is increased. The SPPF module uses the Silu activation

function [23], which improves the accuracy to a certain extent, but also reduces the computation speed and is not suitable for embedded devices. In this paper, a new spatial pyramid pooling module, LR-SPPF, is proposed using Leaky ReLU as the activation function.

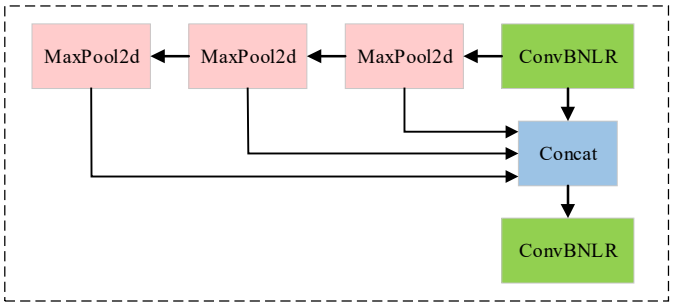

Fig. 5.Schematic diagram of the LR-SPPF structure.

Leaky ReLU not only retains the features of ReLU that make stochastic gradient descent converge faster through sparse network structure. It also adds constant values when the input is less than 0, making the output become negative with small gradient descent. The network structure of the LR-SPPF module is shown in Fig. 5. While retaining the advantages of the SPPF module, the computational speed and applicability are improved, making it more suitable for use in embedded devices.

C. *Neck Improvement*

Neck mainly contains the C3 and Conv modules. The GSConv module was first introduced to reduce the complexity of the model while maintaining a high level of accuracy. In Backbone, the spatial information of the feature map is converted into channel information, so that the information is not complete when it is passed to the Neck. In addition, in order to increase the speed of the model, Neck usually uses sparse convolution rather than dense convolution, a practice that can further lead to loss of information. To solve the above problem, the GSConv module shown in Fig. 6 employs a channel rearrangement trick to rearrange the output of the normal convolution, preserving as much as possible the hidden connections between each channel. The method is able to significantly reduce computational costs when employing sparse convolution and plays an active role in preserving semantic information.

In order to achieve accuracy gains while reducing parameters, this paper proposes several different cross-scale connection optimization methods for the Neck framework, drawing on ideas from the Bidirectional Feature Pyramid Network. First all nodes in the top-down path of the network discard the input features of this layer and select features from the upper layer that have more fused feature information. Considering that some of the gear faults are small targets, which are easily overlooked in the context of complex environments, the fusion of feature information from the original input enhances the prediction of small and medium targets. We designed the I-BiFPN network shown in Fig. 7. 1 is the high-level feature semantic information transfer path. 2 is the low-level feature location information transfer path. 3 is the original input information transfer path.

For practical feature fusion, the simple and efficient weighted feature fusion mechanism Fast Normalized Fusion is used. This mechanism uses a special mathematical expression to determine the weights when fusing feature maps, i.e. by normalizing the mean and standard deviation of the input

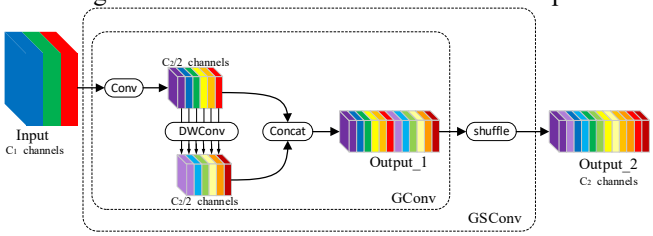

Fig. 6.Schematic diagram of the GSConv network structure.

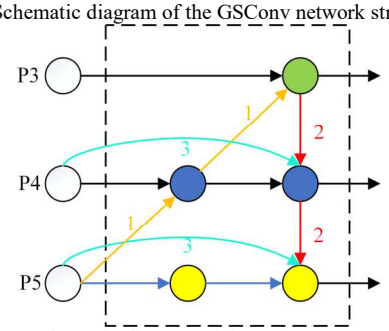

Fig. 7.Schematic diagram of the I-BiFPN structure.

feature maps and then using them as weighting factors to achieve a weighted fusion of the input feature maps. This method is not only simple to use but also offers significant improvements in the effectiveness and speed of feature fusion. Using the P4 layer output as an example, the expression is calculated as:

$$P_4^{td} = \mathrm{Conv}\left(\frac{w_1 \cdot P_4^{in} + w_2 \cdot \mathrm{Resize}(P_5^{in})}{w_1 + w_2 + \grave{o}}\right) \quad (1)$$

$$P_4^{out} = \mathrm{Conv}\left(\frac{w_1' \cdot P_4^{in} + w_2' \cdot P_4^{td} + w_3' \cdot \mathrm{Resize}(P_3^{out})}{w_1' + w_2' + w_3' + \grave{o}}\right) \quad (2)$$

In the formula, P_4^{td} denotes the input; P_4^{out} denotes the output; Resize is the down sampling or up sampling operation; w is the learned parameter to distinguish the importance of different features in the feature fusion process.

D. *Attention Mechanism*

Originally applied in the field of machine translation, attentional mechanisms have become an important part of the network structure with the development of artificial intelligence. In the previous section we have used a large number of lightweight operations, which significantly increase speed but also lead to a decrease in accuracy. To enhance the recognition rate of gear faults, we introduced a variety of attention mechanisms.

The Shuffle Attention structure, shown in Fig. 8, splits channels into a large number of sub-channel groups, uses the Shuffle Unit to construct spatial and channel feature dependencies, and disrupts individual channels, thereby enhancing the exchange of information between features.

Similar to shuffleNetv2, Shuffle Attention also uses the channel shuffle operation to enable cross-group information flow along the channel dimension. This approach is very suitable for the embedded end deployment studied in this paper as it meets the performance requirements while effectively reducing the overall computational cost. Use this module to compensate for the loss of information in the Backbone section caused by the lightening operation.

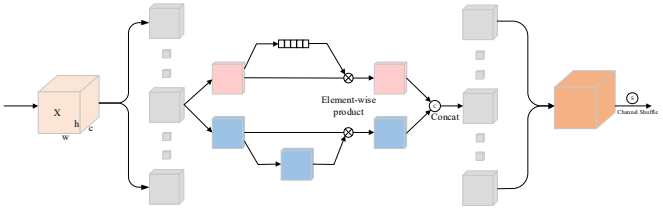

Fig. 8.Schematic diagram of the Shuffle Attention model structure.

The second attention mechanism introduced in this paper is the NAM. based on the integration of the CBAM module, the channel and spatial attention sub-modules are redesigned, and for the channel attention sub-module, a scaling factor is used to represent the size of the individual channel changes. A regularization term is also added to the loss function in order to suppress unimportant features. NAM uses a sparse weight penalty on the attention module, making these weights more computationally efficient while maintaining performance.

IV. Experimental Results and Analysis

A. Datasets and Experimental Platforms

The dataset used in this paper is from a laboratory ammunition loading system experimental stand, which was self-calibrated by camera image acquisition. A total of 2816 images of gear faults were collected at different angles, under different lighting conditions and in different operating conditions. Using LabelImg software the location of the fault information in the dataset was labelled to generate xml files and the dataset was set to PASCAL VOC format. The labels of the dataset images include Miss, One-third miss, Wear, and Bulge. The training and test datasets were randomly divided in a 9:1 ratio for this experiment. Gear failure images covering the categories shown in Fig. 9.

The experimental platform in this paper is divided into two parts, one is the platform used for model training: NVIDIA RTX3080 GPU, AMD EPYC 7601 CPU, CUDA version 11.1, python version 3.8. The second is the platform used for real-time fault detection: Rockchip RK3568 development platform, ubuntu version 20.0, python version 3.8.

B. Assessment Indicators

The ability of the model to detect gear faults in real time in embedded devices was evaluated using model size, computing speed and detection time. Precision, recall, and average precision were used as evaluation metrics to assess the algorithm's recognition.

Precision is used to indicate the percentage of samples with a positive correct prediction out of all samples with a positive prediction.

$$P = \frac{TP}{TP + FP} \times 100\% \tag{3}$$

In the formula: TP indicates the number of positive samples with correct predictions. FP indicates the number of negative samples with positive predictions.

Recall is used to indicate the proportion of samples with positive correct predictions out of the positive samples in the whole data set.

Fig. 9.(a) Gear failure with missing tooth.(b) One-third of gears missing tooth failure.(c) Gear wear failure.(d) Gear bulge failure.

$$R = \frac{TP}{TP + FN} \times 100\% \tag{4}$$

In the formula: FN indicates the number of positive samples that were incorrectly predicted.

AP measures how well a category is detected by calculating the area under the PR curve. The mAP is obtained by averaging Ap, which is a measure of the effectiveness of detection for multiple categories.

$$mAP = \frac{\sum_{k=1}^{N} P(k)\Delta R(k)}{C} \tag{5}$$

In the formula: N is the number of samples in the test set, P(k) is the magnitude of the precision rate when k samples are identified simultaneously, is the change in recall when the number of samples detected changes from k-1 to k, and C is the number of categories.

C. Results Analysis

Fig. 11 shows the prediction results obtained using the LG-YOLOv5 model for detection on the RK3568. The values next to all identifiers are the category and confidence scores.

The confusion matrix of Fig. 10 shows that the LG-YOLOv5 detects all four types of gear faults with an accuracy of 96% or more and 99.8% for Bulge, so the LG-YOLOv5 has good detection capability for gear faults in the munitions loading system.

Using the same dataset, the mainstream detection models faster_rcnn, retinaNet, ssd, YOLOv5 were evaluated separately. From TABLE I, it can be seen that compared to faster_rcnn-MobileNetV2, faster_rcnn-resnet50, retinaNet, ssd, and YOLOv5, LG-YOLOv5 has 12%, 3.1%, 3.6%, 5.1%, and 0.5% higher accuracy and model size respectively reduced by 98.6%, 97.2%, 96.5%, 91.5%, 38.2%. As you can see, the

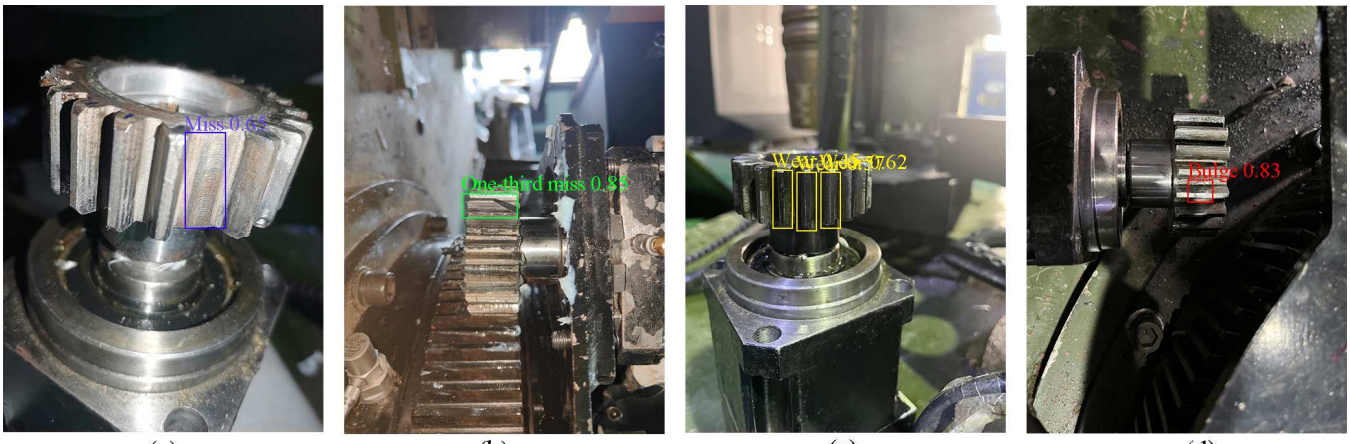

Fig. 11. (a) Miss, confidence level 0.65. (b) One-third miss, confidence level 0.85. (c)Wear, confidence level 0.62.(d)Bulge, confidence level 0.83.

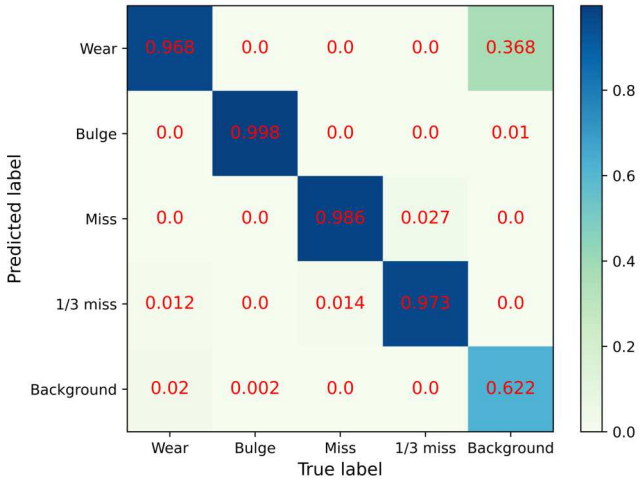

Fig. 10.Confusion matrix.

LG-YOLOv5 not only has a significantly reduced model size, but also maintains a high level of accuracy.

To evaluate the new model more fully, we deployed both the LG-YOLOv5 model and the YOLOv5 model to the RK3568 for testing, as shown in TABLE II. The overall accuracy of the model is reduced compared to that on the RTX3080 due to the limited computing power on the board side. For model comparison, the LG-YOLOv5 offers a 1.5% improvement in board-side accuracy and a 0.5% improvement in recall over the YOLOv5. In TABLE III, the LG-YOLOv5 has 45% faster single image calculations, 55.2% faster video stream detection, 39.8% less parameters and 86.3% less floating point operations. It proves that the LG-YOLOv5 proposed in this paper performs well on the embedded platform and can effectively improve the efficiency and accuracy of gear fault detection.

D. Ablation Study

In this section, we verify the impact of the improvements proposed in this paper on fault detection by means of ablation experiments. Different evaluation metrics are used for different modules. ShuffleNetV2 is primarily intended to be lightweight and is therefore evaluated using model size, number of operations and detection time. I-BIFPN and Attentional mechanism are used to maintain accuracy using precision rate, average precision.

The parameters in TABLE IV represent the experimental results with the relevant modules removed. It can be seen that with the removal of ShuffleNetV2, the model volume increased by 69.3% and the detection time increased by 169%, proving that the module effectively reduces the overall size of the model and is more suitable for embedded end deployment.

Removing the I-BIFPN and Attentional mechanism reduced the prediction precision by 1% and 1.9%, proving that the module was effective in improving the model's ability to detect faults.

V. Conclusion

The LG-YOLOv5 algorithm proposed in this paper is a lightweight gear fault detection algorithm. The main advantages of the algorithm are: (1) a significant reduction in the number of model parameters and model size while maintaining accuracy; and (2) accurate and fast identification of gear faults in an embedded platform with low computing power. To achieve a lightweight model, we used a variety of technical means. Firstly, we achieved a significant reduction in computational effort by replacing the YOLOv5 backbone network with a ShuffleNetV2 lightweight network, thereby reducing the number of parameters. Secondly, the GSConv module was introduced in the neck section to further lighten the model. Then we propose the LR-SPPF module, which improves the BiFPN structure and enables multi-layer feature fusion by adding multiple paths for more complete feature

TABLE I
Comparison of LG-YOLOV5 with Mainstream Detection Models

Method	P (%)	mAP (%)	Weight Size (MB)
faster_rcnn-MobileNetV2	86.1	94.8	629.3
faster_rcnn-resnet50+FPN	95.0	96.3	315.1
retinaNet	94.5	95.1	248.2
ssd	93.0	95.0	103.1
YOLOv5	97.6	98.8	14.3
LG-YOLOv5	98.1	98.9	8.8

TABLE II
LG-YOLOV5 VS. YOLOV5 PRECISION RELATED PARAMETERS

Method	P (%)	R (%)	mAP (%)	Weight Size (MB)
YOLOv5	96.3	99	98.2	14.3
LG-YOLOv5	97.8	99.5	98.4	8.8

TABLE III
LG-YOLOV5 VS. YOLOV5 DETECTION EFFICIENCY

Method	T1 (image /s)	T2(video /s)	FLOPs (G)
YOLOv5	0.9123	1.110	16.0
LG-YOLOv5	0.5017	0.497	2.2

TABLE IV
RELEVANT PARAMETERS FOR FAULT DETECTION WITH THE RK3568

Method	P (%)	mAP (%)	FLOPs (G)	Weight Size (MB)	T(s)
I-BIFPN	96.8	97.6	2.1	8.5	0.503
ShuffleNetV2	98	99.6	16.3	14.9	1.337
Attention	95.9	97.2	2.2	8.8	0.483
LG-YOLOv5	97.8	98.4	2.2	8.8	0.497

information transfer. Finally, a mixture of Shuffle Attention and NAM Attention Mechanism modules allows the model to focus more on the region of interest, which improves the extraction of gear fault features.

To verify the effectiveness of LG-YOLOv5 for gear fault detection, we ported it to the Rockchip RK3568 development platform. The LG-YOLOv5 model has 38.2% less volume than the YOLOv5 model, 86.3% fewer FLOPs, 1.5% better detection accuracy and 45% better processing time per image. We have also conducted a large number of comparison and ablation studies, and the experimental results show that LG-YOLOv5 has the best overall performance compared to other mainstream models, with the advantages of high detection accuracy, small training models and high speed. It has great potential in gear fault detection and can provide an efficient solution for gear fault detection.

REFERENCES

[1] Sreepradha, C., Krishna Kumari, A., Elaya Perumal, A. et al. "Neural network model for condition monitoring of wear and film thickness in a gearbox. " Neural Comput & Applic 24, 1943–1952 (2014).

[2] Liu, Xiao Hui; Xu, Yong Gang; Guo, De Ying; Liu, Fei (2014). "Mill Gear Box of Intelligent DiagnoLG Based on Support Vector Machine Parameters Optimization. " Applied Mechanics and Materials, 697(), 239–243.

[3] M. Cerrada, G. Zurita, D. Cabrera, R.-V. Sánchez, M. Artés, and C. Li, "Fault diagnoLG in spur gears based on genetic algorithm and random forest," Mechanical Systems and Signal Processing, vol. 70–71, pp. 87–103, Mar. 2016.

[4] M. Zhao, M. Kang, B. Tang, and M. Pecht, "Deep Residual Networks With Dynamically Weighted Wavelet Coefficients for Fault DiagnoLG of Planetary Gearboxes," IEEE Trans. Ind. Electron., vol. 65, no. 5, pp. 4290–4300, May 2018.

[5] L. Yu, Z. Wang, and Z. Duan, "Detecting Gear Surface Defects Using Background-Weakening Method and Convolutional Neural Network," Journal of Sensors, vol. 2019, pp. 1–13, Nov. 2019.

[6] D. Xi, Y. Qin, and S. Wang, "YDRSNet: an integrated YOLOv5-Deeplabv3 + real-time segmentation network for gear pitting measurement," J Intell Manuf, Nov. 2021.

[7] Z. Zheng, J. Zhao, and Y. Li, "Research on Detecting Bearing-Cover Defects Based on Improved YOLOv3," IEEE Access, vol. 9, pp. 10304–10315, 2021.

[8] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design," in Computer Vision – ECCV 2018, vol. 11218, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 122–138.

[9] H. Li, J. Li, H. Wei, Z. Liu, Z. Zhan, and Q. Ren, "Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles".

[10] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and Efficient Object Detection," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, Jun. 2020, pp. 10778–10787.

[11] Q.-L. Zhang and Y.-B. Yang, "SA-Net: Shuffle Attention for Deep Convolutional Neural Networks," in ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, Jun. 2021, pp. 2235–2239.

[12] Y. Liu, Z. Shao, Y. Teng, and N. Hoffmann, "NAM: Normalization-based Attention Module." arXiv, Nov. 24, 2021. Accessed: Mar. 02, 2023.

[13] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios," in 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, Oct. 2021, pp. 2778–2788.

[14] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 779–788.

[15] GitHub. YOLOV5-Master. 2021. Available online: https://github.com/ultralytics/YOLOv5.git/ (accessed on 1 March 2021).

[16] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection." arXiv, Apr. 22, 2020. Accessed: Mar. 02, 2023.

[17] C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, Jun. 2020, pp. 1571–1580.

[18] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," AAAI, vol. 34, no. 07, pp. 12993–13000, Apr. 2020.

[19] J. Ren, Z. Wang, Y. Zhang, and L. Liao, "YOLOv5-R: lightweight real-time detection based on improved YOLOv5," J. Electron. Imag., vol. 31, no. 03, Jun. 2022.

[20] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size." arXiv, Nov. 04, 2016. Accessed: Mar. 02, 2023.

[21] A. Howard et al., "Searching for MobileNetV3," in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), Oct. 2019, pp. 1314–1324.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[23] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical Evaluation of Rectified Activations in Convolutional Network." arXiv, Nov. 27, 2015. Accessed: Mar. 02, 2023.