

## Article

# YOLOv8-LMG: An Improved Bearing Defect Detection Algorithm Based on YOLOv8

Minggao Liu <sup>1</sup>, Ming Zhang <sup>2</sup>, Xinlan Chen <sup>1</sup>, Chunting Zheng <sup>1</sup> and Haifeng Wang <sup>2,\*</sup> 

<sup>1</sup> School of Energy and Mining Engineering, Shandong University of Science and Technology, Qingdao 266590, China; 202283010090@sdust.edu.cn (M.L.)

<sup>2</sup> School of Information Science and Engineering, Linyi University, Linyi 276002, China; zhangming@lyu.edu.cn

\* Correspondence: wanghaifeng@lyu.edu.cn; Tel.: +86-13563912296

**Abstract:** In industrial manufacturing, bearings are crucial for machinery stability and safety. Undetected wear or cracks can lead to severe operational and financial setbacks. Thus, accurately identifying bearing defects is essential for maintaining production safety and equipment reliability. This research introduces an improved bearing defect detection model, YOLOv8-LMG, which is based on the YOLOv8n framework and incorporates four innovative technologies: the VanillaNet backbone network, the Lion optimizer, the CFP-EVC module, and the Shape-IoU loss function. These enhancements significantly increase detection efficiency and accuracy. YOLOv8-LMG achieves a mAP@0.5 of 86.5% and a mAP@0.5–0.95 of 57.0% on the test dataset, surpassing the original YOLOv8n model while maintaining low computational complexity. Experimental results reveal that the YOLOv8-LMG model boosts accuracy and efficiency in bearing defect detection, showcasing its significant potential and practical value in advancing industrial inspection technologies.

**Keywords:** bearing defect; automatic detection; VanillaNet; Shape-IoU; Lion optimizer; CFP-EVC



**Citation:** Liu, M.; Zhang, M.; Chen, X.; Zheng, C.; Wang, H. YOLOv8-LMG: An Improved Bearing Defect Detection Algorithm Based on YOLOv8. *Processes* **2024**, *12*, 930. <https://doi.org/10.3390/pr12050930>

Academic Editors: Shaoke Wan, Naipeng Li and Zijian Qiao

Received: 2 April 2024  
Revised: 23 April 2024  
Accepted: 28 April 2024  
Published: 2 May 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Bearings play a crucial role in chemical equipment and are the core components that support the rotating shaft. The performance of bearings directly affects the operating stability and reliability of equipment. Poor bearing performance will cause the imbalance of rotating parts, resulting in increased vibration and noise of the equipment. It may even lead to equipment failure, shutdown, or damage, seriously affecting the stable operation and reliability of the equipment. Bearing failure can also lead to the separation of rotating parts from equipment, resulting in mechanical hazards and potentially causing serious accidents, injury, and property damage. In addition, maintenance and the replacement of bearings require equipment to be suspended, which also reduces production efficiency. Therefore, bearings are essential to ensure the smooth progress of the chemical production process. The quality of bearings may be affected by various defects during production, assembly, and transportation, such as grooves, wear, scratches, etc. These defects may pose a risk to the regular operation of the equipment, so the detection of bearing defects is crucial. Traditional defect detection methods, such as visual inspection and simple sensor-based methods, can detect problems to a certain extent. Still, these methods are often inefficient and struggle to meet the needs of high-precision and real-time monitoring. Advancements in computer vision and AI provide new solutions for detecting bearing defects. These modern detection methods have the characteristics of high intelligence, high accuracy, and high efficiency, and they bring breakthroughs in the field of defect detection.

At present, two-stage object detection algorithms (such as R-CNN [1], Fast R-CNN [2], and Faster R-CNN [3]) and single-stage object detection algorithms (such as the YOLO [4] series) have been widely used in defect detection. The two-stage algorithm first generates candidate regions, then classifies these regions, and performs boundary box regression.

Although the accuracy is high, the speed is slow, and it is challenging to meet the needs of real-time detection. In contrast, a single-stage algorithm (such as YOLOv8 [5]) directly predicts the category and location of the target from the image, eliminating the step of generating candidate regions, so it can achieve faster detection speed while ensuring high accuracy, and, as such, is more suitable for real-time monitoring applications.

As an advanced single-stage target detection algorithm, YOLOv8 can effectively identify various types of defects in bearings through its unique network architecture and optimization technology, including those subtle defects in complex backgrounds. This algorithm not only improves the accuracy of detection but also significantly improves the detection speed and provides an efficient and accurate technical solution for detecting bearing defects. Therefore, a new bearing defect detection algorithm based on the YOLOv8 model is proposed in this work. The specific contributions of this work are as follows:

- (1) A novel bearing defect detection model has been developed, leveraging VanillaNet as its core network to enhance its capability in identifying subtle defects on the bearing surface. This approach simplifies the network architecture, significantly reducing model complexity and computational cost. The Lion optimizer was adopted to accelerate the training process further and enhance detection accuracy. It suits intricate defect detection tasks, improving efficiency by ensuring effective data utilization and rapid model convergence.
- (2) Integrating the CFP-EVC module has significantly enhanced the ability of the model to identify complex, occluded, and overlapping defects. The advanced feature fusion and enhanced strategy optimization networks have led to faster processing speeds. Moreover, introducing the Shape-IoU loss function has improved the position accuracy of the model, which is particularly useful for detecting minor defects and providing more precise detection boundary evaluation.
- (3) Extensive experimental verification was carried out on the bearing defect dataset collected by chemical enterprises. Compared with the current mainstream target detection models, the proposed method improves the detection accuracy and significantly reduces the computational resources required. The experimental results also prove that the model is robust and practicable in practical industrial applications.

In the face of these challenges, the need for more effective detection methods is crucial. This research introduces a groundbreaking approach to detecting bearing defects through the YOLOv8-LMG model. This model builds upon the strengths of the YOLOv8n framework, enhancing it with the VanillaNet backbone network for more robust feature extraction, the Lion optimizer for faster convergence, the CFP-EVC module to handle complex, overlapping defects, and the Shape-IoU loss function for more accurate defect localization. These innovations improve detection accuracy, efficiency, and speed, enabling real-time and reliable monitoring of bearing conditions. The proposed model, YOLOv8-LMG, significantly enhances our ability to detect subtle and critical defects that could jeopardize the stability and safety of chemical production processes. Our model achieves high precision with a mean average precision (mAP) of 86.5% at an intersection over union (IoU) of 0.5 and 57.0% at IoU 0.5–0.95, outperforming existing models while maintaining low computational demands. These capabilities make YOLOv8-LMG a pivotal advancement in industrial defect detection, ensuring the safety and efficiency of equipment critical to the chemical industry.

This is divided into the following sections: Section 2 provides an overview of the current state of research in the field of defect recognition, both domestically and internationally. Section 3 primarily introduces YOLOv8 and the methods developed in this work to enhance it, referred to as YOLOv8-lmg. Section 4 focuses on various experiments and their results. Section 5 describes the conclusions and prospects of this work.

## 2. Related Work

### 2.1. Traditional Bearing Defect Recognition

The field of bearing defect detection has long relied on a range of traditional methods, including but not limited to vibration analysis, acoustic emission monitoring, thermal imaging, and eddy current testing. Each method provides a sensitive and accurate diagnosis of defects that may occur during bearing operation. Recent technological advancements have offered new perspectives for fault diagnosis in mechanical systems. Specifically, various detection technologies have demonstrated their strengths and potential in diagnosing bearing defects.

Integrating infrared thermal imaging technology with deep learning has introduced innovative methods for bearing fault diagnosis. Shao [6,7] et al. have applied infrared thermal imaging technology combined with an improved convolutional neural network (CNN) for fault diagnosis under variable operating conditions of the rotor-bearing system. This method not only showcases the application of thermal imaging technology in detecting bearing defects but also demonstrates how deep learning technology can enhance the performance of traditional diagnostic methods. Additionally, findings by Choudhary [8] et al. indicate that infrared thermal imaging technology can automatically identify bearing faults in a non-contact manner, facilitating early detection and warnings and reducing system downtime.

Traditional wear particle and vibration analysis technologies remain essential tools in fault diagnosis. Lin [9] et al. conducted an in-depth analysis of wear particles in railway bearings, revealing the wear mechanisms under rolling/sliding contact and adhesion. Abdeltwab [10] et al. have reviewed the application of vibration analysis in engine fault diagnosis, emphasizing the importance of denoising techniques, such as higher-order statistics and wavelet transforms. Subsequently, Liu [11] employed an empirical wavelet thresholding method to analyze the vibration signals of large-scale wind turbine blade bearings, which are particularly susceptible to weak fault signals due to their size and operational speeds combined with environmental noise interference. Their methodology effectively isolated fault signals from complex datasets, demonstrating the applicability of this technique within nonlinear dynamic systems. Finally, Hou [12] et al., through a comparative research of vibration monitoring and acoustic emission (AE) technology, optimized the fault diagnosis method for high-speed train wheelset bearings, providing a natural transition to the introduction of acoustic emission analysis technology.

Applying acoustic emission (AE) technology in bearing fault diagnosis has gradually gained attention. Research on acoustic emissions by Liu [13] has provided strong technical support for diagnosing faults in wind turbine blade bearings and high-speed train axle bearings, demonstrating the effectiveness of AE technology in monitoring micro-cracks and early-stage faults.

Eddy current testing technology has unique advantages in crack detection and material performance evaluation. Research by Zhang [14] has showcased the application of eddy current technology in crack detection and the design of high-speed permanent magnet machines. Yu [15] et al. introduced a multi-objective optimization method for three-degree-of-freedom hybrid magnetic bearings, considering eddy current effects and saturation issues. They proposed a dynamic magnetic circuit model and a design method for maximum carrying capacity and minimum cost, demonstrating the effectiveness of the optimization results.

Deep learning overcomes the limitations of traditional diagnostic methods, such as low sensitivity, reliance on manual interpretation, and limited early detection capabilities. Deep learning algorithms can automatically interpret data to improve fault detection accuracy and identify complex fault patterns in varying conditions. This marks a significant advancement in diagnostic methods towards higher precision, efficiency, and reliability, paving the way for the development of bearing fault diagnostic technology.

## 2.2. Detection Methods Based on Deep Learning

In the field of bearing defect identification, the contributions of deep learning, notably reviewed by Moshayedi [16], highlight its transformative impact across various applications, including advanced model abstractions and nonlinear transformations within large databases. Specifically, the application achievements of deep learning, especially the YOLO algorithm, continue to emerge. Fu [17] et al., by combining the improved YOLOv5 and K-Means++ algorithm, significantly improved the accuracy and efficiency of detection. This demonstrates the great potential of deep learning technology in practical industrial applications. Following this, Merainani [18] et al. used an innovative approach combining infrared thermal vision technology with the YOLO-v4 framework to provide a new perspective for automatically detecting hot-bearing boxes on rails. In addition, the improvement of YOLOv3 by Zheng [19] et al. and the YOLOv5 model enhanced by gamma transform by Zhao [20] et al. have all contributed to the early detection and accurate identification of bearing defects. In particular, the further optimization of the YOLOv5 network by Xu [21] et al. not only improved the feature extraction capability of the model but also enhanced its diversity and robustness. This series of research work jointly promoted the development of bearing defect identification technology.

With the advancement of technology, researchers are now paying attention to applying YOLOv8 in identifying bearing defects using deep learning models. The application of the YOLOv8s model in the agricultural field by Yang [22] et al. demonstrated the cross-domain potential of deep learning technology, while the improved YOLOv8 algorithm proposed by Wen [23] et al. for crop leaf disease detection achieved remarkable results in terms of balancing detection accuracy and model weight. Xiong [24] et al. optimized the accuracy of bridge floor crack detection through the YOLOv8-GAM-Wise-IoU model; Zhang [25] et al.'s DsP-YOLO optimization in small-size defect detection and Cao [26] et al.'s application in photovoltaic defect detection are also notable. These studies show the application value of YOLOv8 in bearing defect identification and reflect the broad applicability and powerful performance of deep learning technology in different fields.

Identifying bearing defects in chemical enterprises is challenging. YOLOv8 excels in general object detection but falls short in detecting specific bearing defects in chemical environments, resulting in lower accuracy and efficiency. To address this, we have implicitly improved the model for this application scenario: Firstly, VanillaNet [27] was adopted as the backbone network, which has been optimized for the characteristics of bearings in chemical enterprises.

This optimization enhances the ability to capture subtle defect features, aligning with the requirements for scientific research documentation. Simultaneously, the Lion optimizer [28] was introduced to accelerate model training, significantly improving training efficiency, especially considering the complexity and diversity of data in a chemical environment. Furthermore, by integrating the CFP-EVC [29] module, our model achieved significant improvements in identifying complex defects in bearings in chemical enterprises, especially in detecting invisible defects, such as early damage and minor cracks. Finally, we optimized the localization accuracy of the model using the Shape-IoU [30] loss function, ensuring accurate defect identification and precise localization of the defect location, providing vital information for subsequent repair and replacement.

This series of targeted improvements enhances the model performance in identifying bearing defects in chemical enterprises and accelerates the speed of model training and deployment while ensuring high recognition accuracy. This is significant for improving the efficiency of chemical enterprises, reducing maintenance costs, and preventing potential equipment failures. As such, it will enhance the application value of our work.

### 3. Algorithm

#### 3.1. YOLOv8

In January 2023, Ultralytics, the team behind YOLOv5, released its latest object detection framework, YOLOv8. YOLOv8 has excelled in classification, detection, segmentation, and attitude estimation, but its specific workings remain undisclosed. A significant improvement of YOLOv8 is the anchor-free design, which improves the detection speed and enhances the model accuracy. In addition, YOLOv8 has some upgrades and enhancements over previous versions, introducing new features to improve the model’s performance and flexibility.

YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x are available in multiple versions to suit different application needs, all of which outperform previous YOLO versions on the COCO dataset. Considering the model size, YOLOv8n was chosen as the research focus. The architecture of YOLOv8 is divided into four main parts: the input, backbone, neck, and head. At the input, the model accepts images with a size of  $640 \times 640$ . The backbone is partly based on YOLOv5 but significant improvements have been made, such as changing the first layer volume from  $6 \times 6$  to  $3 \times 3$ , replacing the C3 module with the C2f module, and taking a page from the YOLOv7ELAN design. In the neck part, the sampling layer of  $1 \times 1$  convolution is eliminated, and the C2f module is leveraged to enhance feature extraction. The head section implements a decoupled design, separating classification and regression tasks. Mosaic data enhancement was used to preprocess images and improve model generalization. Mosaic enhancement was caused in the last 10 epochs to optimize detection precision. The model structure is shown in Figure 1.

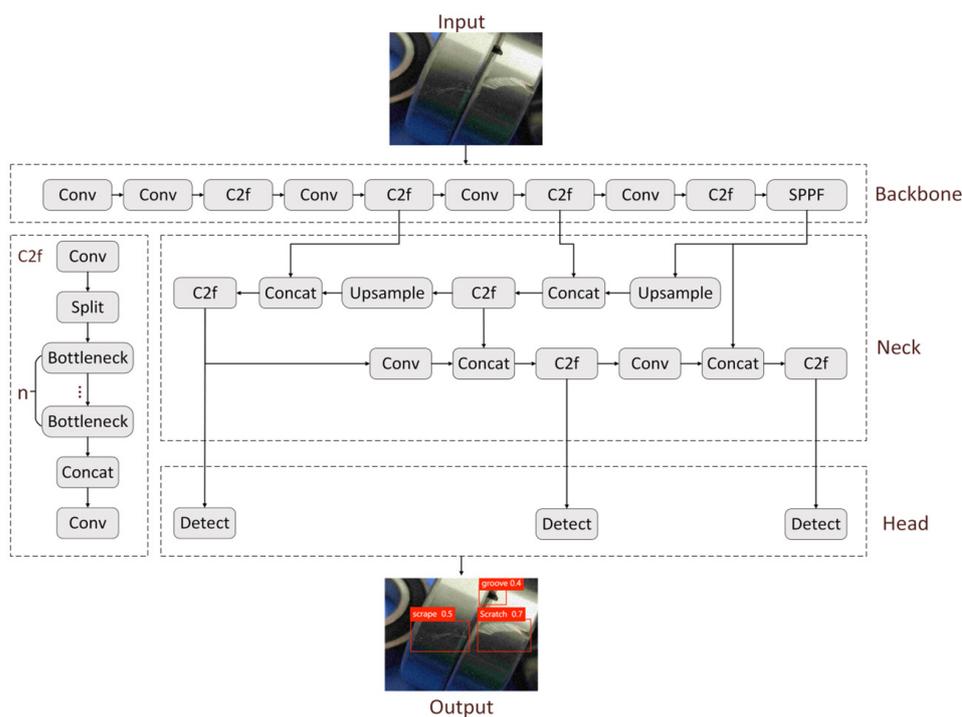
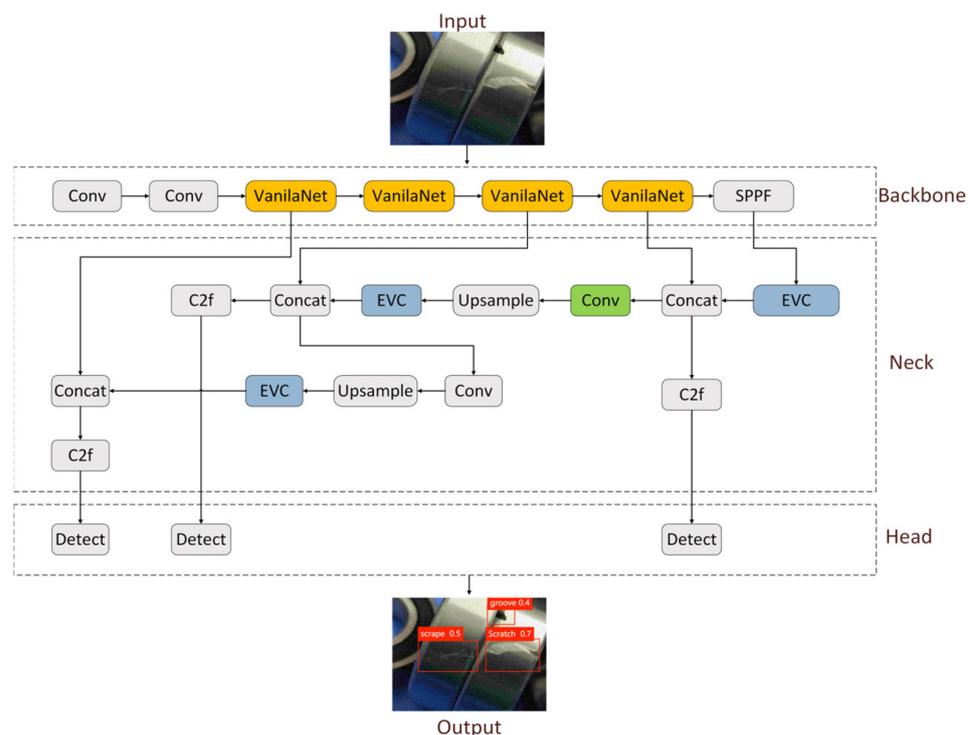


Figure 1. YOLOv8n network framework.

The Conv module of YOLOv8 consists of Conv2d, batch normalization (BN), and the SiLU activation function, which is the core component of the model, and which is responsible for extracting and processing feature information. The C2f module combines the design of the C3 module with the lightweight attention mechanism of ELAN to maintain the efficiency and performance of the model. Another critical component is serial parallel pooling fusion (SPPF), which is based on the concept of spatial pyramid pooling (SPP) to expand the receptive field by concatenating multiple maximum pooling layers while reducing the number of parameters and the computational burden.

### 3.2. YOLOv8-LMG

Bearings, vulnerable to damage, exhibit many defect types that complicate detection. The YOLO8 model, despite its broad applicability, faces difficulties in detecting the nuanced and complex characteristics of bearing defects. This work introduces an advanced detection model, enhancing defect identification capabilities beyond the constraints of YOLOv8. This model, enriched with precise optimization strategies and sophisticated algorithms, significantly improves detection accuracy and operational efficiency. Using deep learning has facilitated identifying bearing defects in industrial inspection, enhancing production efficiency and superior quality. This technological advancement has a direct impact on the production process. Based on this, this research aims to make innovative improvements to the YOLOv8 model to enhance its ability to identify bearing defects in complex environments. For ease of identification, it was denoted as YOLOv8-LMG. The YOLOv8 model had low accuracy and missed minor defects, especially in samples with complex backgrounds. Therefore, this work will analyze these problems and present the corresponding solutions. VanillaNet, a lightweight and efficient network structure, is introduced to replace the original YOLOv8 backbone network to improve the ability of the model to extract bearing defect features. Meanwhile, to adapt to the characteristics of the bearing defect dataset, the Lion optimizer developed by Google was introduced in this research. Through its advanced adaptive learning rate adjustment mechanism, the training process of the model was further accelerated, and the recognition accuracy was improved. A centralized feature pyramid network EVC (CFP-EVC) module has been added to improve the ability of the model to detect bearing defects. This module enhances the detection performance of the mold by effectively fusing and centrally processing features. The performance is more significant when dealing with defects with large size changes, occlusion, and complex backgrounds. In terms of loss function, the shape-IoU loss function was proposed in this research to replace the traditional IOU loss function to more accurately evaluate the ability of the model to recognize the shape of bearing defects, thereby improving the overall detection accuracy. The improved model structure is shown in Figure 2.



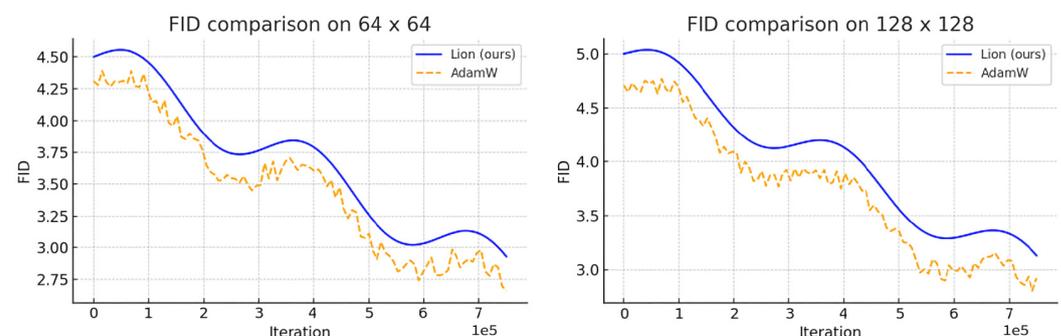
**Figure 2.** YOLOv8-LMG network framework.

Figure 2 shows that the original convolutional layers in the backbone of YOLOv8 have been replaced with the VanillaNet network. This choice is because VanillaNet is proficient in extracting complex patterns from visual data, which is suitable for identifying complex and subtle features of bearing defects that the original architecture may not capture effectively. In the neck area of YOLOv8-lmg, EVC modules have been incorporated to process multi-scale feature maps created by the backbone. These modules optimize feature information by providing enhanced contextual and visual cues, enabling the model to differentiate and defect features more accurately. The design of the YOLOv8 framework aims to address the difficulties that the YOLOv8 algorithm faces in recognizing complex and minute defects in bearing defect detection within chemical enterprises. The improvements brought by the VanillaNet and EVC modules overcome the limitations of the original YOLOv8 model, delivering greater defect detection accuracy and a deeper understanding of complex defect features. Although the Shape-IoU loss function and Lion optimizer play algorithmic roles during the training phase and are not depicted in the architecture flow, they complement the structural enhancements by fine-tuning the ability of the model to localize and accelerate the learning process. The principles of these components and their role in enhancing model performance will be detailed later in the text.

### 3.2.1. Google Lion Optimizer

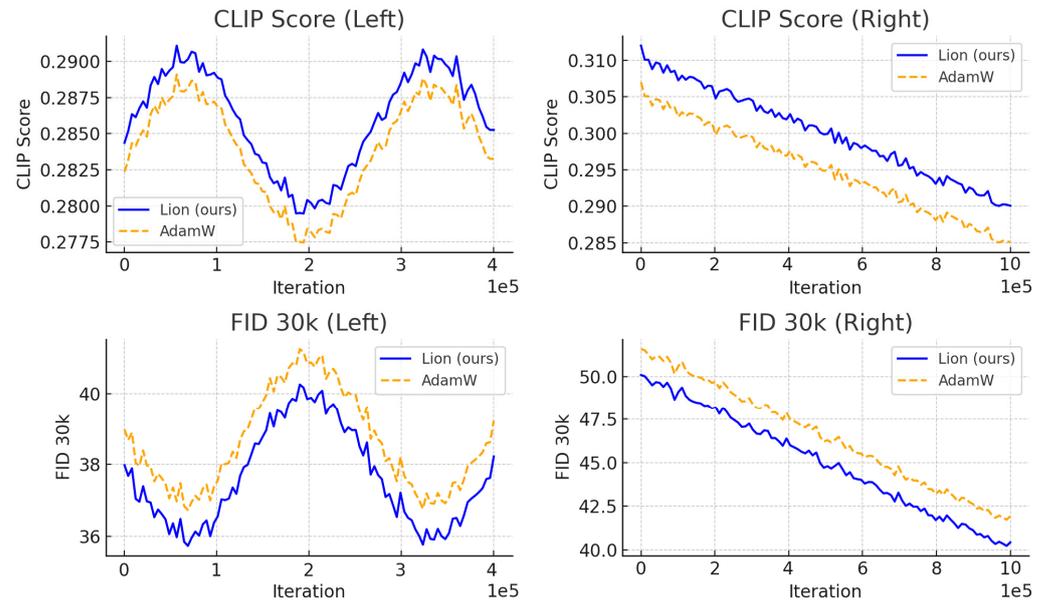
Based on a further understanding of Google Lion optimizer, we used this novel optimization algorithm to analyze the bearing defect datasets. The Lion optimizer was developed through algorithm discovery and program search techniques, focusing on the efficiency and effectiveness of deep neural network training. It uses an efficient search strategy to explore significant and sparse program spaces and to narrow the generalization gap between agent and target tasks through program selection and simplification strategies. The main innovation of Lion relies only on momentum tracking, which is more memory efficient than other adaptive optimizers, such as Adam. In addition, Lion calculates updates through symbolic operations, assigning the same number of updates to each parameter, simplifying the optimization process and improving the accuracy and efficiency of training.

Figure 3 shows that the Lion optimizer exhibits superior quality and quicker convergence regarding Fréchet inception distance (FID) scores across image generation tasks at resolutions of  $64 \times 64$  and  $128 \times 128$ . The prowess of the Lion optimizer becomes increasingly evident in higher-resolution tasks, signifying its robustness in managing more intricate challenges. For instance, in the generation of  $128 \times 128$  images, as the iterations progress, there is a noticeable decline in FID scores for the Lion optimizer, indicating an enhanced quality of image generation within the same number of steps compared to the AdamW optimizer. This efficiency gain is crucial for high-resolution image tasks, which typically demand more computational resources and time. Hence, the characteristics demonstrated by the Lion optimizer could be significantly beneficial for improving the training efficiency of large-scale image generation models.



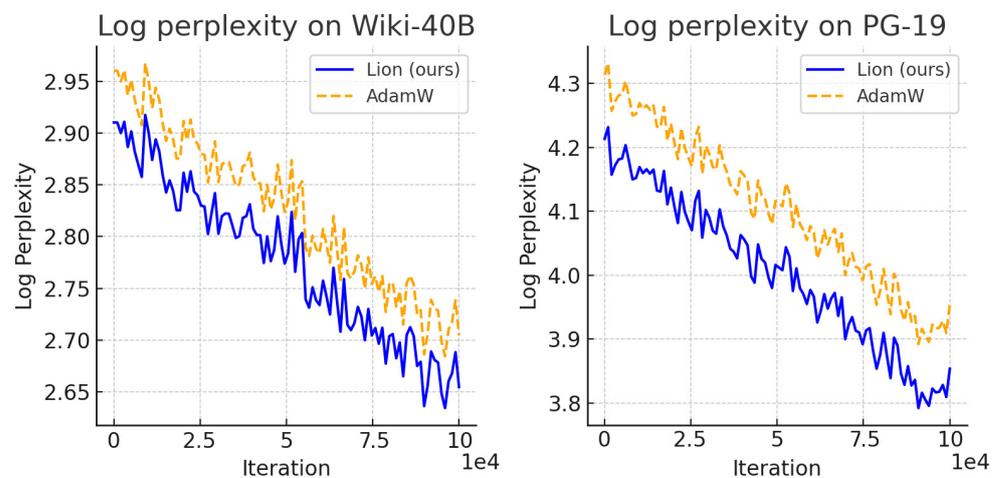
**Figure 3.** FID comparison on  $64 \times 64$  (Left) and  $128 \times 128$  (Right) image generation when training diffusion models.

In Figure 4, the Lion optimizer outperforms AdamW on the Imagen text-to-image super-resolution model, achieving a higher CLIP score and lower FID indicator volatility. These results demonstrate the advantages of the Lion optimizer in improving image text alignment and image realism.



**Figure 4.** Evaluation of the Imagen text-to-image 64<sup>2</sup> (Left) and the 64<sup>2</sup> → 256<sup>2</sup> diffusion models (Right).

Figure 5 shows the log perplexity performance of the Lion optimizer on the Wiki-40B dataset and the performance on the PG-19 dataset. Logarithmic confusion is a commonly used performance index in natural language processing, which is used to measure the quality of language model prediction samples. The lower the value, the more accurate the prediction of the model. The graph shows that the acceleration effect of the Lion optimizer becomes more evident with the increase in model size. The largest model was omitted on the Wiki-40B dataset due to severe overfitting. This suggests that Lion may have a more significant performance improvement than other optimizers when dealing with large datasets, especially when overfitting is needed.



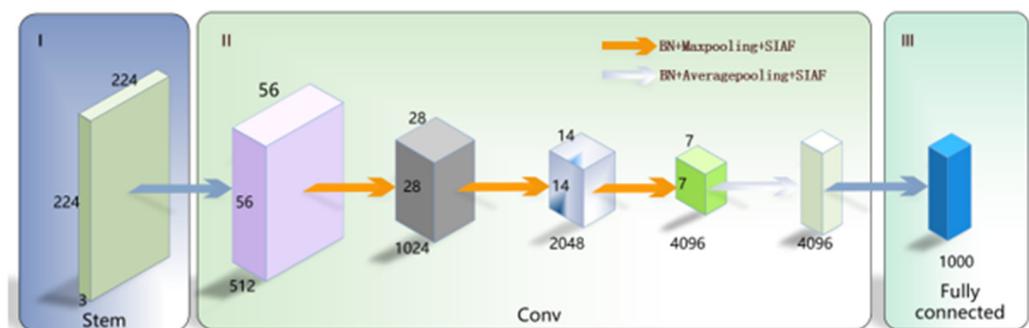
**Figure 5.** Log perplexity on Wiki-40B (Left) and PG-19 (Right).

Taking this information together, the Lion optimizer shows significant performance improvements over the AdamW optimizer in image synthesis (Figure 3), text-to-image super-resolution generation (Figure 4), and large-scale language modeling (Figure 5). In some tasks, such as bearing defect detection, the uniform update volume and built-in regularization effects of Lion may help to improve the ability of the model to generalize on unseen data and reduce the risk of overfitting while maintaining training efficiency. Therefore, these characteristics of the Lion optimizer indicate that it can achieve fast and accurate model training and better detection performance in high-precision applications, such as bearing defect detection.

### 3.2.2. VanillaNet Backbone Network

VanillaNet represents an attempt to move towards simplicity and design elegance, taking a basic but efficient approach to neural network architecture design. Its core philosophy is “diversity means difference”, a principle that has seen notable success in certain fields, such as computer vision and natural language processing. Nevertheless, the inherent complexity of the optimization process and the Transformer model presents new challenges that have prompted researchers to search for simpler design paradigms. The design of VanillaNet avoids deep network structures, shortcuts, and complex operations, such as self-attention mechanisms, seeking to maintain powerful performance through simplification. After training, this design simplifies each layer and reverts to the original architecture by trimming the non-linear activation function, reducing model complexity.

The VanillaNet network architecture is shown in Figure 6. The architecture includes the following elements:



**Figure 6.** VanillaNet network framework.

This diagram illustrates the network architecture of VanillaNet, which is structured into three principal sections:

I. Stem: This initial segment handles preliminary feature extraction through convolutional layers, processing an input image size of 224x224 pixels with three channels, indicative of the RGB color space.

II. Conv: Representing the convolutional stages of the network, this crucial segment is tasked with feature extraction and learning. The varying dimensions of feature maps signal processing at distinct layers, with numbers denoting the spatial dimensions and depth (number of channels or features), such as 1024 and 2048.

III. Fully connected: This final section consists of fully connected layers responsible for the classification or other relevant tasks based on the learned features. It translates the outputs from the preceding layers into 1000 units, typically correlating to 1000 different classes for tasks like image classification.

The arrows indicate data flow and the interconnections between layers. Additionally, the diagram delineates the integration of different pooling strategies (Maxpooling and Averagpooling) with batch normalization (BN) and specific activation functions (SIAP), depicting their respective impact within the architecture. The architecture includes the following elements:

1. The input is represented as a three-dimensional block, suggesting an image with a height and width of 224 pixels and 3 color channels.
2. The network comprises several convolutional layers, as indicated by the smaller three-dimensional blocks where the spatial dimensions (height and width) are reduced while the depth increases. These layers are responsible for feature extraction. Each layer is followed by a pooling layer, which further reduces the spatial dimensions (height and width), as shown by the decrease in the size of successive blocks.
3. After multiple convolutional and pooling layers, the representation becomes much deeper (indicated by the increased depth of the blocks) but with reduced spatial dimensions.
4. Towards the end of the network, the architecture seems to include fully connected layers, represented by flat, elongated rectangles. These layers typically interpret the features extracted by the convolutional layers and make decisions based on them.
5. The final part of the network shows a transition from a fully connected layer with 4096 units to an output layer with 1000 units. This suggests that the network is designed for a classification task with 1000 possible categories.
6. Arrows indicate the direction of data flow from the input to the output.

This simplified approach by VanillaNet is particularly suitable for resource-limited environments because it overcomes the challenges of complexity while maintaining the effectiveness of the model. Simple architecture enables efficient deployment of deep learning models on devices with limited computing resources. Extensive experimental results indicate that although the design of VanillaNet is highly simplified, its performance is comparable to that of current deep neural networks and vision Transformer models. This demonstrates the efficacy of implementing a minimalist approach in deep learning.

### 3.2.3. CFP-EVC

The Centralized Feature Pyramid with Explicit Visual Centers (CFP-EVC) module is designed to solve some limitations of the existing visual feature pyramid methods in target detection. These methods often focus on the interaction of features between layers and neglect the importance of intra-layer feature conditioning. However, the latter has been empirically shown to be beneficial for improving the model’s performance. Attempts to enhance in-layer feature representation by introducing attentional mechanisms or visual converters often leave out corner regions of the input image that are critical for intensive prediction tasks. The model diagram is shown in Figure 7.

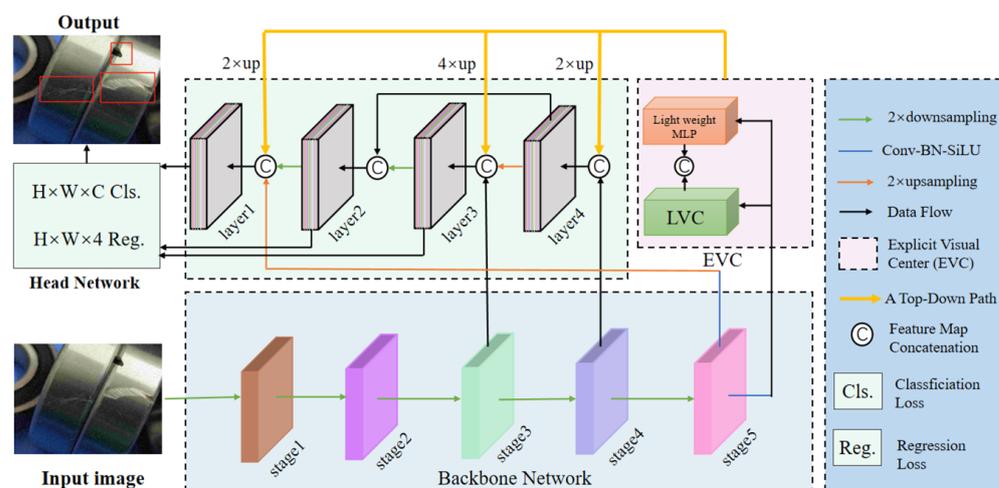


Figure 7. CFP-EVC model framework.

The given image displays a neural network object detection architecture that starts with an input image processed through a backbone network with multiple stages (stage 1 to stage 5) for feature extraction. The head network follows, employing upsampling and concatenation to integrate multi-scale features for object classification and localization. Specialized components, such as lightweight MLP, LVC, and EVC modules enhance network contextual understanding and focus on informative image regions. The CFP-EVC module aims to overcome this problem by explicitly centralizing feature conditioning globally, thereby optimizing the use of feature pyramids in target detection. It captures global remote dependencies through a spatially explicit vision center scheme that uses a lightweight multi-layer perceptron (MLP). In addition, the CFP-EVC module utilizes a learnable visual center mechanism in parallel that is specifically targeted to capture local corner regions of the input image, thus ensuring that the model takes full advantage of all critical information in the image.

CFP-EVC implements the top-down global centralized regulation strategy based on this design idea. This strategy utilizes explicit visual center information from the deepest feature to effectively adjust the forward shallow feature to enhance the feature expression ability of each layer in the feature tower. CFP-EVC can efficiently capture global remote dependencies and obtain comprehensive feature representations, improving target detection performance.

### 3.2.4. Shape-IoU

Shape-IoU (shape intersection ratio) is an innovative border regression loss calculation method to improve positioning accuracy in object detection tasks. Traditional frame regression loss calculation methods mainly focus on the relative position and shape relationship between the predicted frame and the ground truth (GT), but often ignore the influence of the shape and scale properties of the frame itself on the positioning precision. The proposed Shape-IoU method aims to fill this research gap and optimize the process of border regression by considering the inherent properties of the border shape and scale.

The core idea of Shape-IoU is to incorporate the shape and scale factors of the border itself into the loss calculation. In this way, Shape-IoU focuses not only on the relative relationship between the GT frame and the prediction frame but also on the geometric properties of the prediction frame itself. This method can guide the model in carrying out border regression more accurately to improve the localization accuracy of target detection. Shape-IoU calculates the ratio of intersection and union between the predicted target shape and the real target shape. The calculation method of Shape-IoU can be summarized as follows:

1. Calculate IoU (intersection over union)

The formula for calculating the standard IoU is as follows:

$$IoU = \frac{|b \cap b^{gt}|}{|b \cup b^{gt}|} \quad (1)$$

where  $b$  and  $b^{gt}$  represent the predicted box and the ground truth box, respectively.

Introducing the scale factors for weighted width  $ww$  and weighted height  $hh$  is calculated as follows:

$$ww = \frac{2 \times (w^{gt})^{scale}}{(w^{gt})^{scale} + (h^{gt})^{scale}} \quad (2)$$

$$hh = \frac{2 \times (h^{gt})^{scale}}{(w^{gt})^{scale} + (h^{gt})^{scale}} \quad (3)$$

2. Calculate shape distance.

The specific calculation formula is outlined as follows:

$$distance^{shape} = hh \times (x_c - x_c^{gt})^2 / c^2 + ww \times (y_c - y_c^{gt})^2 / c^2 \quad (4)$$

3. Calculate the shape consistency term obtained by cumulatively computing weighted width and height differences. The exponential decay function here is used to evaluate shape consistency.

The following formula defines the precise computation:

$$\Omega^{shape} = \sum_{t=w,h} (1 - e^{-wt})^{\Theta}, \Theta = 4 \quad (5)$$

where  $w_w$  and  $w_h$  are the proportions of weighted differences in width and height, calculated by the following formula:

$$\begin{cases} w_w = hh \times \frac{|w - w^{gt}|}{\max(w, w^{gt})} \\ w_h = ww \times \frac{|h - h^{gt}|}{\max(h, h^{gt})} \end{cases} \quad (6)$$

4. Calculate Shape-IoU

The exact mathematical expression for the calculation is as follows:

$$\mathcal{L}_{Shape-IoU} = 1 - IoU + distance^{shape} + 0.5 \times \Omega^{shape} \quad (7)$$

Shape-IoU is particularly suitable for applications requiring the accurate assessment of object shape matches, such as segmentation tasks, 3D reconstruction, and some types of object detection, providing an intuitive way to measure the similarity between predicted and real shapes.

## 4. Experiment and Analysis

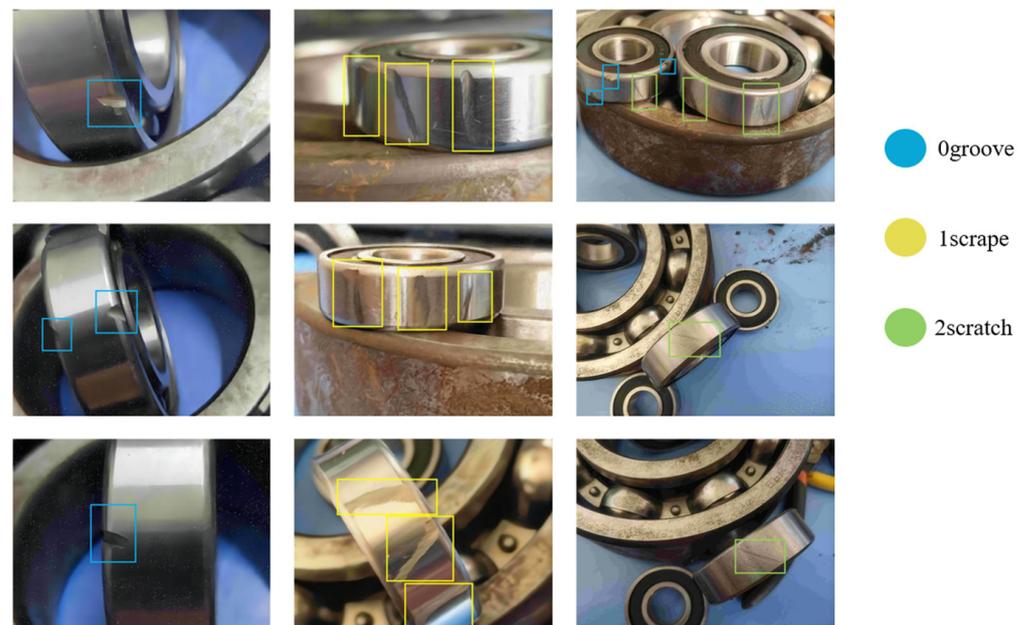
### 4.1. Datasets and Evaluation Indicators

This research collected and utilized a pre-use defect detection dataset of chemical plant equipment bearings, which included 6543 images to capture the possible defects of bearings during different stages of production, assembly, and transportation. The datasets were divided into training sets, verification sets, and test sets with a ratio of 8:1:1. All images have been annotated in detail, and each image corresponds to a text file that records the specific location and type of defect, providing the necessary input information for the training, validation, and testing of the model.

Considering the diversity in and uncertainty of bearing defects, this dataset includes a variety of defect types, including groove defects, scratch defects, and scrape defects. The types of defects are shown in Figure 8, where 0 represents “groove”, 1 represents “scrape”, and 2 represents “scratch”. These defects have different effects on the performance of bearings, so it is particularly critical to carry out accurate detection and classification in practical applications. At the same time, the diversity of defect features in the dataset, such as shape, size, and location, put forward high generalization ability and robustness requirements for the flaw detection algorithm to adapt to the wide range of defect variation in the real scene.

Four key indicators were used to evaluate the model performance: mean accuracy (mAP), frames processed per second (FPS), number of model parameters (Params), and floating-point operations (GFLOPs). mAP measures the average accuracy of the model in detecting various targets on the test set. The higher the value, the more powerful the detection ability of the model. The FPS indicator reflects the image processing speed of the model, and the higher the FPS, the faster the response speed of the model and the better the real-time performance. Parameter number (Params) is an indicator to evaluate the size and

complexity of a model, and the fewer parameters, the more concise the model structure. GFLOPs is used to measure the computing resources required during the execution of the model, and a lower GFLOPs value indicates that the model has higher computational efficiency. These four indexes together constitute a comprehensive framework to measure the performance of the model in terms of precision, speed, and efficiency, which can fully reflect the performance of the model in practical applications.



**Figure 8.** Types of defects diagram.

#### 4.2. Experimental Settings

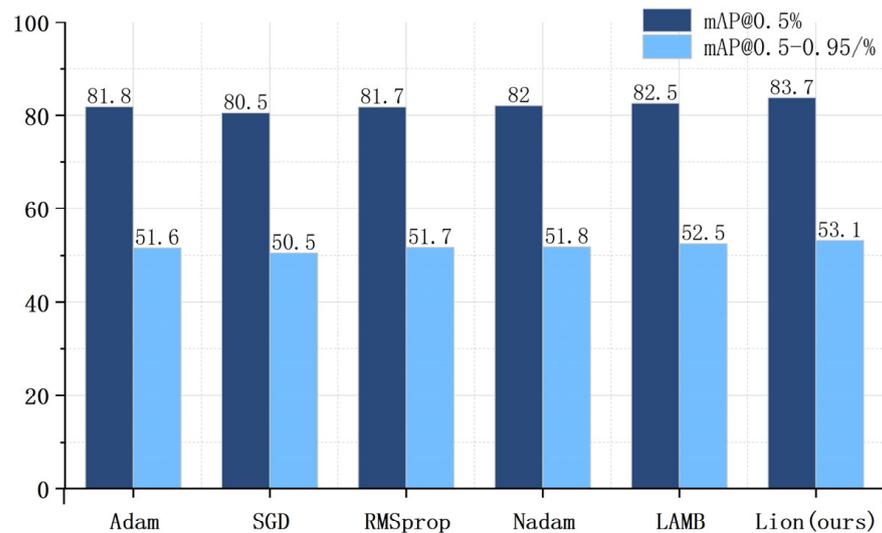
The hardware configuration includes an NVIDIA Tesla V100 and an Intel(R) Xeon(R) Silver 4210 CPU 2.20 GHz; The GPU is configured with 16 GB HBM2 (High Bandwidth Memory 2). In terms of software, this work uses Python3.7, PyTorch 1.7.0, and Cuda11.3 deep learning frameworks. Using the Google Lion optimizer, the experiment set the initial learning rate to 0.01, the batch size to 32, the epoch to 200, and the input image size to  $640 \times 640$ .

#### 4.3. Comparison Experiment

##### 4.3.1. Comparative Analysis of Lion Optimizer

This work compares the Google Lion optimizer with other state-of-the-art optimizers, such as Adam, SGD, etc. The comparison chart is shown in Figure 9. The Lion optimizer scored 83.7% on mAP@0.5, 1.9 percentage points higher than the standard Adam optimizer. More significantly, the Lion optimizer scored 53.1% in mAP@0.5–0.95, 1.5 percentage points higher than Adam, 2.6 percentage points higher than SGD, and 1.4 percentage points higher than RMSprop. It was 1.3 percentage points higher than Nadam and 0.6 percentage points higher than LAMB.

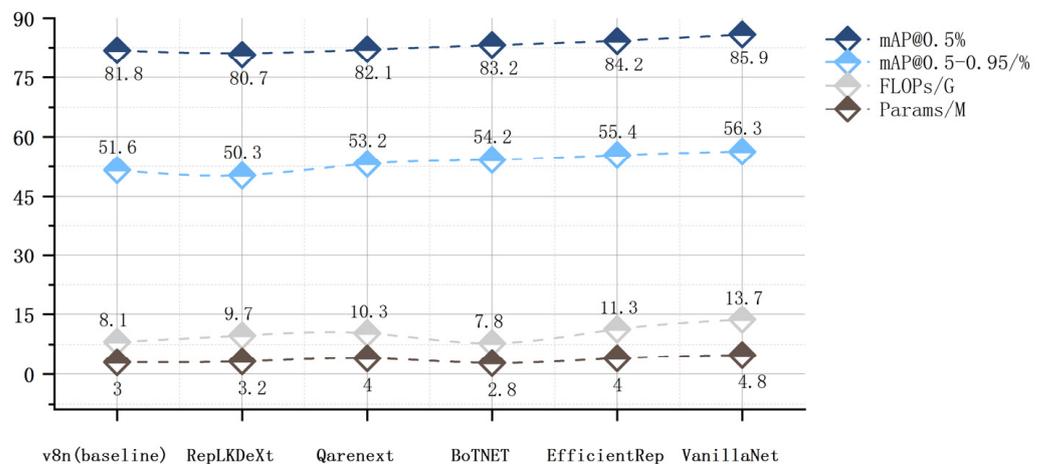
These results show that the Lion optimizer provides the best accuracy in bearing defect identification, especially in mAP@0.5–0.95 evaluations, considering a wider range of IoU thresholds. While FLOPs and the number of parameters remained the same for all optimizers (8.1 G and 3.0 M), the performance improvement with the Lion optimizer demonstrated its efficiency in optimizing model parameters. Therefore, the Lion optimizer has apparent advantages for improving the accuracy of bearing defect detection, and there is no increase in computational cost, which makes the Lion optimizer a superior choice for bearing defect detection tasks.



**Figure 9.** The comparison chart of the Google optimizer with other optimizers.

#### 4.3.2. Comparative Analysis of VanillaNet

As can be seen from Figure 10, VanillaNet showed excellent results in key performance indicators when comparing different backbone network models. Specifically, in the mAP@0.5 evaluation, VanillaNet led with an accuracy of 85.9%, which is 4.1 percentage points higher than the baseline model YOLOv8 at 81.8%. In the more rigorous mAP@0.5–0.95 evaluation, VanillaNet outperformed the other models with an average accuracy of 56.3%, 4.7 percentage points higher than YOLOv8. This result highlights the advantages of VanillaNet in precisely identifying targets. However, this high accuracy comes at the cost of increased computational complexity, with VanillaNet having a FLOPs/G value of 13.7, much higher than other models, such as BotNET with a FLOPs/G value of 7.8, indicating that the computational requirements of VanillaNet are almost twice those of BotNET. In terms of the number of parameters, VanillaNet has 4.8 M parameters, which is second only to EfficientRep and Qarenext, each with 4.0 M parameters. Although VanillaNet is not optimal regarding the number of parameters and computational complexity, it provides the highest accuracy on object detection tasks, especially on more fine-grained IoU evaluation criteria. This shows that VanillaNet is a superior choice for identifying bearing defects.



**Figure 10.** The comparison chart of the performance between VanillaNet and other backbone networks.

#### 4.3.3. Comparative Analysis of CFP-EVC

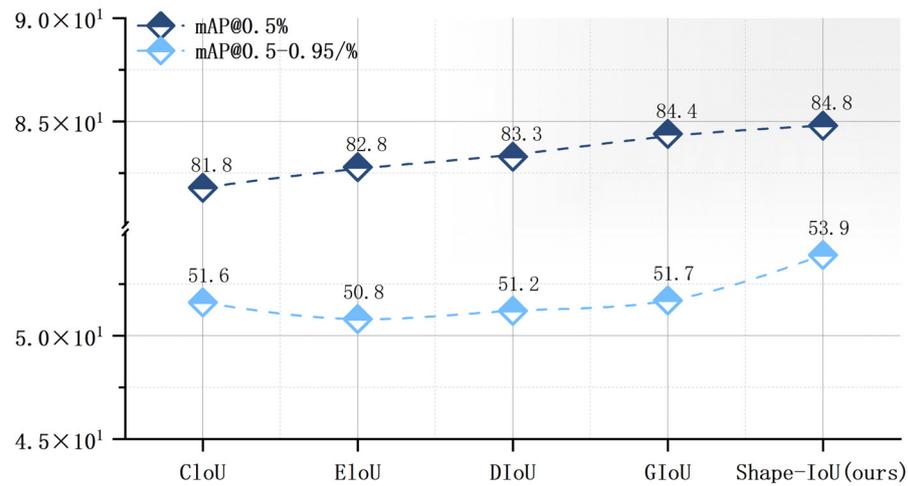
It can be seen from Figure 11 that YOLOv8-C2-CFP-EVC shows significant advantages in the task of bearing defect identification compared with other models. Figure 11 displays an  $x$ -axis with four performance metrics for object detection models: mean average precision at IoU 0.5 (mAP@0.5) and IoU 0.5–0.95 (mAP@0.5–0.95) in percentages, reflecting accuracy; floating-point operations per second in billions (FLOPs/G), indicating computational efficiency; and model parameters in millions (Params/M), representing size and complexity. While plotted on the same axis, the units—percent for mAP, billions for FLOPs, and millions for Params—differ, with higher mAP denoting better accuracy, lower FLOPs suggesting efficiency, and fewer Params implying a more streamlined model. Our model achieves 84.6% on mAP@0.5, 2.8 centenaries higher than the baseline model YOLOv8n. In the more fine-grained mAP@0.5–0.95 index, YOLOv8-C2-CFP-EVC scored 55.3%, which is far higher than other models and 3.7 percentage points higher than the baseline model, indicating that our model has more accurate recognition ability under different IoU thresholds. Even compared to YOLOv8-C2-Repghost, our model has a 2.2% improvement on mAP@0.5–0.95. In terms of computational complexity (FLOPs/G), although YOLOv8-C2-CFP-EVC is not the lowest, its value of 7.7 is still on the low side, only 0.4 higher than the optimal YOLOv8-C2-GhostNet. Regarding model size (Params/M), the number of parameters of YOLOv8-C2-CFP-EVC is 2.7 M. Compared with most other models, YOLOv8-C2-CFP-EVC performs better without a significant parameter increase; it has 0.3 M fewer parameters than YOLOv8n. In summary, choosing YOLOv8-C2-CFP-EVC as the backbone network can provide higher accuracy for bearing defect identification while maintaining reasonable computational and parametric efficiency. This makes the YOLOv8-C2-CFP-EVC model ideal for high-precision target detection in resource-constrained situations.



**Figure 11.** Comparison chart of YOLOv8-C2-CFP-EVC module versus other modules.

#### 4.3.4. Comparative Analysis of Shape-IoU

According to Figure 12, the Shape-IoU model performs better than other models in bearing defect identification tasks. Shape-IoU scored the highest on mAP@0.5, with 84.8%, which is three percentage points higher than CloU, which scored the lowest at 81.8%. More significantly, Shape-IoU reaches 53.9% on mAP@0.5–0.95, 2.3 percentage points higher than the CloU model and 2.2 percentage points higher than the GIoU model, the next highest evaluation index. The mAP@0.5–0.95 metric is critical because it takes into account average performance across IoU thresholds ranging from lax to strict, and is a more comprehensive measure of performance.



**Figure 12.** Performance comparison chart between Shape-IoU and other metrics.

Regarding computational complexity (FLOPs/G) and some parameters (Params/M), Shape-IoU is consistent with other models at 8.1 G and 3.0 M. This means that Shape-IoU provides a significant increase in performance while maintaining the same computational burden as other models. Such performance advantages make Shape-IoU the preferred model for bearing defect identification tasks, especially in resource-constrained environments, which can achieve higher detection accuracy while maintaining low computational complexity.

#### 4.4. Ablation Experiment

To comprehensively evaluate the performance of the YOLOV8-LMG algorithm proposed in this work in the task of bearing defect detection, this research introduced four innovative modules based on the core of the YOLOv8 algorithm: Lion, VanillaNet, CFP-EVC, and Shape-IoU. We designed 12 detailed ablation experiments to rigorously assess the contribution of each module toward enhancing detection accuracy. All experiments were carried out in a unified experimental environment with the same hyper-parameter configuration to ensure the consistency and comparability of the experimental results.

The experimental results are shown in Table 1, where ‘√’ indicates that the module was introduced in the corresponding experiment:

**Table 1.** Ablation experiment results.

Group	Lion	VanillaNet	CFP-EVC	Shape-IoU	mAP @0.5%	mAP@0.5-0.95%	FLOPs/G	Params/M
1					81.8	51.6	8.1	3
2	√				83.7	53.1	8.1	3
3		√			85.9	56.3	13.7	4.8
4			√		84.6	55.3	7.7	2.7
5				√	84.8	53.9	8.1	3
6	√	√			84.7	54.7	8.1	3.4
7	√		√		85.3	55.2	11	3.9
8	√			√	84.8	54	8.1	3
9		√	√		86	55.5	13.7	4.8
10		√		√	85.4	55.1	13.7	4.8
11			√	√	85.2	55.8	7.7	2.7
12	√		√	√	85.5	56	11	3.9
13	√	√		√	85	54.5	8.1	3.4
14	√	√	√		86.2	56.2	11.5	4.1
15	√	√	√	√	86.5	57	13	4.5

For instance, introducing the Lion module in Experiment 2 increased the mAP@0.5 to 83.7% and mAP@0.5–0.95 to 53.1%, with no change in FLOPs or parameters, illustrating the capacity of this module to boost accuracy without additional computational overheads. The results show that the Lion module can effectively improve the measurement accuracy without increasing the calculation burden.

In Experiment 3, after introducing the VanillaNet module, mAP@0.5 increased to 85.9%, mAP@0.5:0.95 increased to 56.3%, although FLOPs increased to 13.7 G, and the number of parameters increased to 4.8 M. This result confirms that VanillaNet dramatically improves the detection performance and correspondingly increases the computational complexity of the model.

In Experiment 4 and Experiment 5, CFP-EVC and Shape-IoU modules were added, respectively, and both of them increased mAP@0.5 and mAP@0.5:0.95 to varying degrees while maintaining low FLOPs and the number of parameters, demonstrating the effectiveness of these modules in enhancing the model performance. This is particularly obvious in improving the accuracy of the model for the detection of bearing defects.

When combining modules, such as Lion with VanillaNet (Experiment 6), Lion with CFP-EVC (Experiment 7), and Lion with Shape-IoU (Experiment 8), the results showed considerable improvements in mAP metrics, validating the synergistic effects of these combinations on enhancing detection capabilities. The synergistic effect of different module combinations is verified to improve the detection performance further.

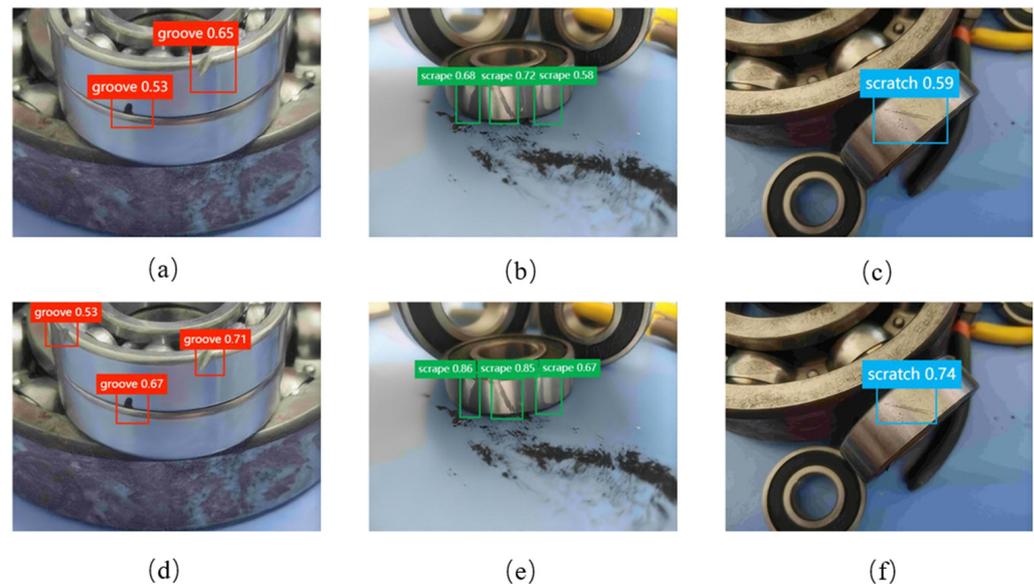
Experiment 12 shows that when all four innovative modules (Lion, VanillaNet, CFP-EVC, and Shape-IoU) are introduced, mAP@0.5 reaches 86.5%, mAP@0.5:0.95 reaches 57.0%, and the FLOPs value is 13.0 G. The number of parameters is 4.5 M. This balanced approach underscores the combined efficiency and accuracy of the modules and demonstrates the capability of the model to optimize performance while managing computational complexity and parameter count effectively. Through the above ablation experiments, this research not only verified the effectiveness of various innovation points in improving the detection performance of the YOLOv8-lmg algorithm, but also demonstrated how to significantly improve the detection accuracy of the algorithm while reducing the number of parameters through the comprehensive application of these innovative technologies in the task of bearing defect detection. It provides strong technical support for practical application scenarios.

#### 4.5. Qualitative Analysis

In this experiment, we compared the performance of the improved target detection algorithm and the original YOLOv8 algorithm on the bearing defect dataset. The resolution of the input image in the experiment is  $640 \times 640$ , and the confidence threshold is set at 0.25. Through qualitative analysis, we find that the improved algorithm is superior to YOLOv8 regarding detection accuracy and reliability. The comparison results between the enhanced algorithm and the YOLOv8n algorithm are shown in Figure 13, where (a), (b), and (c) represent the improvements of YOLOv8, and (d), (e), and (f) represent the improvements of YOLOv8-LMG.

In relatively explicit picture scenes, the YOLOv8 algorithm shows a certain degree of missing and false detection phenomenon, especially in recognizing grooves and scratches, which is not accurate enough. In contrast, the improved algorithm can more accurately locate and identify the bearing grooves, and its ability to capture fine features has been significantly improved. In the complex background, especially in recognizing scratch features, YOLOv8 also has shortcomings, with many missed detections. However, the improved algorithm has no apparent missing phenomenon, reflecting its feature extraction and target location improvement.

In the small-target detection scenario, YOLOv8 is not as effective at identifying defects at a distance or small size, which are often not successfully detected. The improved algorithm shows excellent detection ability and can identify small targets missed by YOLOv8, showing advantages in dealing with small-size targets.



**Figure 13.** Comparative analysis of object detection outcomes: YOLOv8 versus YOLOv8-LMG where (a–c) represent the improvements of YOLOv8, and (d–f) represent the improvements of YOLOv8-LMG.

The improved algorithm shows higher accuracy and stability in detecting bearing defects. Its performance in dealing with complex fields and small-size targets is better than the original YOLOv8 algorithm. This result shows that optimizing the original algorithm can effectively improve the feature extraction ability and the detection performance of small targets to achieve more accurate and reliable defect detection in practical applications.

#### 4.6. Compared with Algorithms Utilizing the Same Dataset

To enhance the credibility of the findings and provide a robust benchmarking framework, this section details a comparative analysis between the YOLOv8-LMG and the GRP-YOLOv5 [20] algorithms, both tested using the same bearing defect dataset from a chemical enterprise. This fair and unbiased comparison is pivotal in assessing the relative advancements facilitated by the modifications inherent in the YOLOv8-LMG design. The comparative results of the algorithms are shown in Table 2.

**Table 2.** Comparative Performance Metrics of YOLOv8-LMG and GRP-YOLOv5 on the Chemical Enterprise Bearing Defect Dataset.

Algorithm	Recall	Precision	mAP@0.5	mAP@0.5:0.95	FNR	F-Score
GRP-YOLOv5 [20]	87.4%	93.2%	93.5%	52.7%	12.6%	90.2%
YOLOv8-LMG	89%	93.5%	86.5%	57%	11%	91.2%

The YOLOv8-LMG demonstrated a recall rate of 89% and a precision of 93.5%, which marginally surpasses the performance metrics of GRP-YOLOv5, which achieved a recall of 87.4% and an accuracy of 93.2%. These results underscore the refined capability of YOLOv8-LMG to identify true defects more effectively, a crucial attribute for minimizing the risk of critical failures in chemical manufacturing processes.

Despite the mAP@0.5 for YOLOv8-LMG being 86.5%, slightly lower than the 93.5% for GRP-YOLOv5, this metric reflects targeted refinement of the algorithm for more complex detection scenarios rather than straightforward defect identifications. The mAP@0.5:0.95 significantly increased to 57.0% from 52.7%, indicating robust performance across a comprehensive range of defect sizes and operational conditions. This is especially critical in chemical plants where defect characteristics can vary significantly and are challenging to pinpoint accurately.

Furthermore, the improvement in the false negative rate from 12.6% to 11% and an enhancement in the F-Score from 90.2% to 91.2% highlight improved reliability and a balanced approach towards sensitivity and specificity in YOLOv8-LMG. These enhancements make YOLOv8-LMG a superior choice for high-stakes industrial applications, where defect detection accuracy is paramount and computational efficiency is critically valued.

This comparative analysis reinforces the validity of design improvements in YOLOv8-LMG and significantly showcases its potential to advance defect detection technology in industrial settings. The detailed evaluation against GRP-YOLOv5 using the same dataset lays a solid foundation for deploying YOLOv8-LMG in environments demanding high precision and operational robustness.

#### 4.7. Comparison with Advanced Algorithms

In this research, we conducted a series of target detection algorithm performance comparison tests on the bearing defect dataset for chemical enterprises, aiming to evaluate the performance of the improved YOLOv8 algorithm, which we called YOLOv8-LMG, in practical applications. The comparison algorithms include the two-stage target detection algorithm Fast R-CNN, first-stage target detection algorithms SSD, YOLOv5, YOLOv7, YOLOv8, and CG-Net, and the anchor-free target detection algorithm CenterNet. This comprehensive comparison was designed to accurately assess the position of YOLOv8-lmg among current leading target detection technologies.

The experimental results reveal the significant advantages of YOLOv8-lmg in key performance indicators. Specifically, YOLOv8-lmg reached 86.5% on mAP@0.5%, 57.0% on mAP@0.5–0.95%, 13.0G on FLOPs/G, and 4.5M on the number of model parameters. These results not only show the improvement of YOLOv8-LMG in mAP compared with the original YOLOv8 algorithm (1.8 percentage points), but also show the optimization in the number of model parameters and calculation amount—the number of model parameters is reduced by  $1.05 \times 10^6$ . Table 3, presented below, provides a comparative performance analysis of the YOLOv8-LMG algorithm against other leading algorithms across four performance metrics: mAP@0.5%, mAP@0.5–0.95%, FLOPs, and Params.

**Table 3.** Comparative performance analysis of the YOLOv8-LMG algorithm.

Algorithm	mAP@0.5%	mAP@0.5–0.95%	FLOPs/G	Params/M
YOLOv8n	81.8	51.6	8.1	3.0
Faster R-CNN	82.0	55.0	20.0	25.0
SSD	75.0	50.0	2.5	6.8
RetinaNet	81.0	56.0	10.0	36.0
YOLOv5	82.5	53.5	12.0	3.8
EfficientDet-D3	83.0	54.4	6.1	12.0
CenterNet	80.0	52.0	19.0	20.0
Mask R-CNN	81.5	55.0	26.0	44.0
YOLOv8-LMG(ours)	86.5	57.0	13.0	4.5

The above table illustrates a comparative evaluation of the YOLOv8-LMG algorithm against other algorithms across four performance metrics: mAP@0.5%, mAP@0.5–0.95%, FLOPs, and Params.

Compared with other advanced algorithms, YOLOv8-LMG performs exceptionally well when detecting bearing defects. For example, compared with the CenterNet algorithm based on the anchor-free method, YOLOv8-lmg improves mAP by 7.4 percentage points, reduces the number of mode parameters by about 97%, reduces the computation by about 91.5%, and increases FPS by 222. This comparison not only highlights the significant advantages of YOLOv8-LMG in terms of accuracy improvement but, more importantly, it has achieved great success in terms of model weight and efficiency improvement.

In summary, the performance of the improved YOLOv8-LMG algorithm on the bearing defect dataset of chemical enterprises is superior to the current advanced target detection

algorithm, especially in terms of model accuracy, volume, computational efficiency, and real-time performance. These results show that the YOLOv8-LMG algorithm is very suitable for practical application scenarios requiring high precision and high-efficiency target detection, especially in an environment with limited computing resources.

## 5. Conclusions

This research enhances the YOLOv8n algorithm to create the YOLOv8-LMG model, significantly improving the accuracy and efficiency of bearing defect detection in industrial settings. Initially, the model incorporates VanillaNet as its backbone network. This integration markedly boosts the characteristic extraction capabilities for bearing defects. Compared to the original YOLOv8n model, VanillaNet helps to increase the mean average precision (mAP) at the intersection over the union (IoU) threshold of 0.5 from 81.8% to 86.5%, enhancing it by nearly 4.7 percentage points. Furthermore, introducing the Lion optimizer is crucial in expediting the training phase and improving model convergence. This optimizer is particularly effective in managing high-resolution imaging tasks, setting a robust foundation for tackling complex defect detection scenarios. Additionally, integrating the CFP-EVC module significantly bolsters the capability of the model to recognize intricate defect features, thus elevating overall detection precision. The adoption of the Shape-IoU loss function further refines accuracy in localizing defects, elevating the mAP from 51.3% to 57.0% across the broader IoU range of 0.5 to 0.95, which translates to an improvement of 5.7 percentage points. YOLOv8-LMG balances computational complexity and parameter count, reducing the computational load to 13 GFLOPs and the parameter count to 4.5 million. These advancements not only demonstrate superior performance in the specific task of bearing defect detection but also underscore considerable potential for practical industrial applications.

**Author Contributions:** Conceptualization: M.L. and H.W.; Methodology: M.Z.; Software: C.Z.; Validation: M.L., X.C. and H.W.; Formal analysis: M.L.; Investigation: X.C.; Resources: X.C.; Data curation: X.C.; Writing—original draft preparation: M.L.; Writing—review and editing: M.L.; Visualization: C.Z.; Supervision: H.W.; Project administration: H.W.; Funding acquisition: H.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Shandong Provincial Natural Science Foundation, China, grant number ZR2023MF090. The APC was funded by Professor Wang Haifeng from Linyi University.

**Data Availability Statement:** Due to privacy or ethical restrictions, the data supporting the reported results cannot be publicly shared.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
2. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
3. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)] [[PubMed](#)]
4. 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, Las Vegas, NV, USA, 26 June–1 July 2016.
5. Reis, D.; Kupec, J.; Hong, J.; Daoudi, A. Real-Time Flying Object Detection with YOLOv8. *arXiv* **2023**, arXiv:2305.09972.
6. Shao, H.; Xia, M.; Han, G.; Zhang, Y.; Wan, J. Intelligent Fault Diagnosis of Rotor-Bearing System Under Varying Working Conditions with Modified Transfer Convolutional Neural Network and Thermal Images. *IEEE Trans. Ind. Inf.* **2021**, *17*, 3488–3496. [[CrossRef](#)]
7. Shao, H.; Li, W.; Xia, M.; Zhang, Y.; Shen, C.; Williams, D.; Kennedy, A.; De Silva, C.W. Fault Diagnosis of a Rotor-Bearing System under Variable Rotating Speeds Using Two-Stage Parameter Transfer and Infrared Thermal Images. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 3524711. [[CrossRef](#)]

8. Choudhary, A.; Mian, T.; Fatima, S. Convolutional Neural Network Based Bearing Fault Diagnosis of Rotating Machine Using Thermal Images. *Measurement* **2021**, *176*, 109196. [[CrossRef](#)]
9. Lin, C.-L.; Meehan, P.A. Morphological and Elemental Analysis of Wear Debris Naturally Formed in Grease Lubricated Railway Axle Bearings. *Wear* **2021**, *484–485*, 203994. [[CrossRef](#)]
10. Abdeltwab, M.M.; Ghazaly, N.M. A Review on Engine Fault Diagnosis through Vibration Analysis. *Int. J. Recent Technol. Mech. Electr. Eng.* **2021**, *9*, 1–6.
11. Liu, Z.; Zhang, L.; Carrasco, J. Vibration Analysis for Large-Scale Wind Turbine Blade Bearing Fault Detection with an Empirical Wavelet Thresholding Method. *Renew. Energy* **2020**, *146*, 99–110. [[CrossRef](#)]
12. Hou, D.; Qi, H.; Luo, H.; Wang, C.; Yang, J. Comparative Study on the Use of Acoustic Emission and Vibration Analyses for the Bearing Fault Diagnosis of High-Speed Trains. *Struct. Health Monit.* **2022**, *21*, 1518–1540. [[CrossRef](#)]
13. Liu, Z.; Wang, X.; Zhang, L. Fault Diagnosis of Industrial Wind Turbine Blade Bearing Using Acoustic Emission Analysis. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 6630–6639. [[CrossRef](#)]
14. Zhang, Y.; Wang, H.; Gerada, C. Rotor Eddy Current Loss and Multiphysics Fields Analysis for a High-Speed Permanent Magnet Machine. *IEEE Trans. Ind. Electron.* **2021**, *68*, 5100–5111. [[CrossRef](#)]
15. Yu, C.; Deng, Z.; Mei, L.; Peng, C.; Cao, X. Multiobjective Optimization of 3-DOF Magnetic Bearing Considering Eddy Current Effects and Saturation. *Mech. Syst. Signal Process.* **2023**, *182*, 109538. [[CrossRef](#)]
16. Moshayedi, A.J.; Roy, A.S.; Kolahdooz, A.; Shuxin, Y. Deep Learning Application Pros and Cons Over Algorithm. *EAI Endorsed Trans. AI Robot.* **2022**, *1*, 1–13. [[CrossRef](#)]
17. Fu, X.; Yang, X.; Zhang, N.; Zhang, R.; Zhang, Z.; Jin, A.; Ye, R.; Zhang, H. Bearing Surface Defect Detection Based on Improved Convolutional Neural Network. *MBE* **2023**, *20*, 12341–12359. [[CrossRef](#)] [[PubMed](#)]
18. Merainani, B.; Toullier, T.; Zongo, B.; Sriranjani, S.; Zanaroli, S.; Guiraud, M.; Dumoulin, J. Toward the Development of Intelligent Wayside Hot Bearings Detector System: Combining the Thermal Vision with the Strength of YOLO-V4. In Proceedings of the 2022 International Conference on Quantitative InfraRed Thermography, Paris, France, 4–8 July 2022; QIRT Council: Paris, France, 2022.
19. Zheng, Z.; Zhao, J.; Li, Y. Research on Detecting Bearing-Cover Defects Based on Improved YOLOv3. *IEEE Access* **2021**, *9*, 10304–10315. [[CrossRef](#)]
20. Zhao, Y.; Chen, B.; Liu, B.; Yu, C.; Wang, L.; Wang, S. GRP-YOLOv5: An Improved Bearing Defect Detection Algorithm Based on YOLOv5. *Sensors* **2023**, *23*, 7437. [[CrossRef](#)] [[PubMed](#)]
21. Xu, H.; Pan, H.; Li, J. Surface Defect Detection of Bearing Rings Based on an Improved YOLOv5 Network. *Sensors* **2023**, *23*, 7443. [[CrossRef](#)] [[PubMed](#)]
22. Yang, S.; Wang, W.; Gao, S.; Deng, Z. Strawberry Ripeness Detection Based on YOLOv8 Algorithm Fused with LW-Swin Transformer. *Comput. Electron. Agric.* **2023**, *215*, 108360. [[CrossRef](#)]
23. Wen, G.; Li, M.; Luo, Y.; Shi, C.; Tan, Y. The Improved YOLOv8 Algorithm Based on EMSPConv and SPE-Head Modules. *Multimed. Tools Appl.* **2024**. [[CrossRef](#)]
24. Xiong, C.; Zayed, T.; Abdelkader, E.M. A Novel YOLOv8-GAM-Wise-IoU Model for Automated Detection of Bridge Surface Cracks. *Constr. Build. Mater.* **2024**, *414*, 135025. [[CrossRef](#)]
25. Zhang, Y.; Zhang, H.; Huang, Q.; Han, Y.; Zhao, M. DsP-YOLO: An anchor-free network with DsPAN for small object detection of multiscale defects. *Expert Syst. Appl.* **2024**, *241*, 122669. [[CrossRef](#)]
26. Cao, Y.; Pang, D.; Zhao, Q.; Yan, Y.; Jiang, Y.; Tian, C.; Wang, F.; Li, J. Improved YOLOv8-GD Deep Learning Model for Defect Detection in Electroluminescence Images of Solar Photovoltaic Modules. *Eng. Appl. Artif. Intell.* **2024**, *131*, 107866. [[CrossRef](#)]
27. Chen, H.; Wang, Y.; Guo, J.; Tao, D. VanillaNet: The Power of Minimalism in Deep Learning. In Proceedings of the 37th Conference on Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023.
28. Chen, X.; Liang, C.; Huang, D.; Real, E.; Wang, K.; Liu, Y.; Pham, H.; Dong, X.; Luong, T.; Hsieh, C.-J.; et al. Symbolic Discovery of Optimization Algorithms. In Proceedings of the 37th Conference on Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023.
29. Quan, Y.; Zhang, D.; Zhang, L.; Tang, J. Centralized Feature Pyramid for Object Detection. *IEEE Trans. Image Process.* **2023**, *32*, 4341–4354. [[CrossRef](#)] [[PubMed](#)]
30. Zhang, H.; Zhang, S. Shape-IoU: More Accurate Metric Considering Bounding Box Shape and Scale. In Proceedings of the Computer Vision and Pattern Recognition, Xiamen, China, 12 January 2024.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.