

Article

# LESM-YOLO: An Improved Aircraft Ducts Defect Detection Model

Runyuan Wen <sup>1</sup>, Yong Yao <sup>1,\*</sup> , Zijian Li <sup>1</sup>, Qiyang Liu <sup>1</sup>, Yijing Wang <sup>1</sup> and Yizhuo Chen <sup>2</sup>

<sup>1</sup> School of Computer Science and Technology, Xidian University, Xi'an 710126, China; wenrunyuan@stu.xidian.edu.cn (R.W.); zjian.li@stu.xidian.edu.cn (Z.L.); 22031212452@stu.xidian.edu.cn (Q.L.); 21031211499@stu.xidian.edu.cn (Y.W.)

<sup>2</sup> Guangzhou Institute of Technology, Xidian University, Guangzhou 510530, China; yizhuo@stu.xidian.edu.cn

\* Correspondence: yaoyong@xidian.edu.cn

**Abstract:** Aircraft ducts play an indispensable role in various systems of an aircraft. The regular inspection and maintenance of aircraft ducts are of great significance for preventing potential failures and ensuring the normal operation of the aircraft. Traditional manual inspection methods are costly and inefficient, especially under low-light conditions. To address these issues, we propose a new defect detection model called LESM-YOLO. In this study, we integrate a lighting enhancement module to improve the accuracy and recognition of the model under low-light conditions. Additionally, to reduce the model's parameter count, we employ space-to-depth convolution, making the model more lightweight and suitable for deployment on edge detection devices. Furthermore, we introduce Mixed Local Channel Attention (MLCA), which balances complexity and accuracy by combining local channel and spatial attention mechanisms, enhancing the overall performance of the model and improving the accuracy and robustness of defect detection. Finally, we compare the proposed model with other existing models to validate the effectiveness of LESM-YOLO. The test results show that our proposed model achieves an mAP of 96.3%, a 5.4% improvement over the original model, while maintaining a detection speed of 138.7, meeting real-time monitoring requirements. The model proposed in this paper provides valuable technical support for the detection of dark defects in aircraft ducts.

**Keywords:** aircraft ducts; YOLOv8; technical diagnostics; defect detection



**Citation:** Wen, R.; Yao, Y.; Li, Z.; Liu, Q.; Wang, Y.; Chen, Y. LESM-YOLO: An Improved Aircraft Ducts Defect Detection Model. *Sensors* **2024**, *24*, 4331. <https://doi.org/10.3390/s24134331>

Academic Editor: Jongmyon Kim

Received: 29 May 2024

Revised: 21 June 2024

Accepted: 1 July 2024

Published: 3 July 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

Aircraft ducts are an important component of normal aircraft operation, and the timely detection and maintenance of aircraft ducts are crucial for safe aircraft operation [1]. Defects in aircraft ducts have a significant impact on their performance, and the toughness, strength, and corrosion resistance of aircraft ducts are important factors to ensure their safety. If cracks, deformations, or corrosion are present in the conduit, these conditions will weaken its toughness and strength, making it more prone to fracture and damage during use. The traditional defect detection of aircraft ducts is mostly based on manual handheld endoscopic inspection. Aircraft ducts are small, and the detection time is insufficient, which leads to workers needing to perform fault detection in a narrow environment, which is time-consuming and labor-intensive, and highly dependent on the workers' prior experience. The detection process has high labor intensity, a low detection efficiency, and a high probability of detection errors.

Currently, defect detection in aircraft ducts faces the following major challenges. Firstly, high-precision detection is crucial. The safety of aircraft ducts is of the utmost importance, so the accuracy requirements for defect detection systems are very high. The system must be able to accurately detect any potential defects or anomalies to avoid potential safety risks [2]. Additionally, the detection process is complicated by diverse environments and backgrounds. Different materials are used for aircraft ducts in different aircrafts, some

of which have reflective properties. If LED light sources are solely incorporated into the collection equipment, scratch defects in the reflective aircraft ducts are likely to be overlooked. Furthermore, during operation, stains are often present, which can easily be confused with defects in the ducts. Another issue is the low quality of the existing data. Since aircraft ducts are small aircraft components, the collected data often suffer from insufficient lighting. Direct detection in low-light environments fails to meet the high-precision defect detection requirements, and the data contain significant noise. Therefore, it is necessary to develop an automatic, efficient, and economical inspection framework for aircraft ducts to reduce reliance on manual detection, decrease the difficulty of the task, detect defects early, prevent catastrophic failures, and minimize maintenance downtime [3].

To address the aforementioned issues and improve the detection of small defects in low-light environments, as well as to enable the lightweight deployment of the model on edge computing devices, we propose a novel object detection framework, Light Enhancement defect detection based on Space-to-depth convolution and Mixed-channel attention (LESM)-YOLO. In the light enhancement module, we utilized shared weights and a layer-by-layer connection approach to optimize the illumination components, thereby improving image exposure. The self-calibration module ensures the stable convergence of results at each stage. Additionally, we optimized the dataset to address the issue of low-quality defect images captured in low-light conditions. We replaced the conventional convolution module in YOLOv8 with the space-to-depth convolution module, which transforms spatial information into depth information. This effectively reduces information loss and preserves more detailed features, thereby enhancing the accuracy of feature extraction. Given the prevalence of small target defects in aircraft duct inspection, substituting traditional stride convolution and pooling layers with this module significantly improves defect detection accuracy while reducing the number of parameters, meeting the requirements for high-precision detection. Finally, we employed a hybrid channel attention mechanism to address the issue of neglected spatial feature information in the existing CBAM [4] attention mechanism. This approach integrates channel and spatial information with only a slight increase in parameter count, effectively dealing with the complex environments and backgrounds in aircraft duct defect detection. By using this attention mechanism, irrelevant modules are suppressed, significantly enhancing detection performance. Comprehensive experiments demonstrate that LESM-YOLO outperforms all comparative models in terms of mAP metrics.

Our contributions can be summarized as follows:

1. By analyzing the challenges in detecting defects in aircraft ducts under low-light conditions, we integrated a light enhancement module. This integration addresses the issue of low-quality defect images captured in low-light environments from a model perspective.
2. By examining the characteristics of existing aircraft duct defects, we replaced the standard convolution modules with SPDCConv modules. This effectively reduces information loss and preserves more detailed defect features.
3. To address the complex environments and backgrounds present in aircraft duct defect detection, we incorporated an MLCA into the neck module, significantly enhancing the model's detection performance.

## 2. Related Work

In recent years, significant progress has been made in the field of object detection due to the emergence of convolutional neural networks and attention mechanisms. Object detection algorithms can be divided into two categories: single-stage object detection algorithms and two-stage object detection algorithms. The common single-level algorithms are SSD [5] and YOLO [6]. The single-stage algorithm treats defect localization and classification as a regression problem, achieving end-to-end detection and a fast detection speed. However, due to class imbalances and other reasons, the accuracy of single-stage algorithms is slightly lower than that of two-stage algorithms. The common two-stage object detection

algorithms are Fast R-CNN [7] and Faster R-CNN [8]. These algorithms first generate a region of interest (ROI), and then classify and locate it in the second stage, resulting in high accuracy but a relatively slow detection speed.

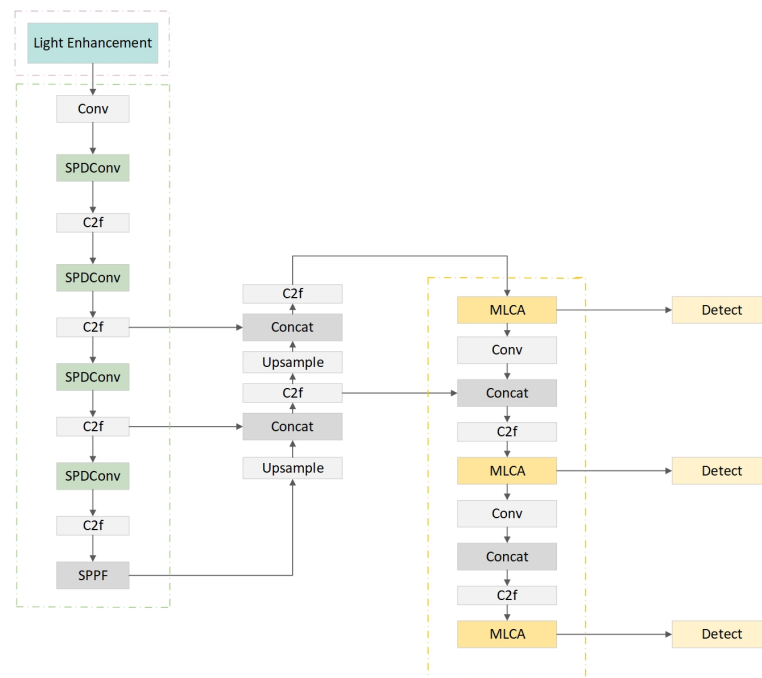
The attention mechanisms in the field of computer science mainly include spatial attention [4,9,10], channel attention [11], dynamic convolution filters, etc. Among them, spatial attention is mainly used to capture image spatial information, such as in object detection, image segmentation, and other tasks [12]. The spatial attention mechanism can help the model focus more on the specific channel information of the image [13]. Channel attention is more commonly applied to tasks that require the processing of image channel information, such as image stylization [14], image super-resolution [15,16], etc. Su et al. [17] utilized channel attention mechanisms to enhance the expressive ability of target features, applying them in the field of object tracking. Dynamic convolution filters are more commonly used in tasks such as multitasking learning [18] and model compression, to help models focus more on the key information of the task. However, spatial attention does not pay enough attention to channel information and cannot achieve optimal results for images with insufficient lighting and low resolution. Channel attention also has the problem of insufficient attention to spatial dimensions. Dynamic convolutional filters have high computational costs and are not suitable for learning single tasks. The above attention mechanisms have certain limitations, which make it difficult to apply them well to the detection of defects in aircraft ducts.

Due to the relatively narrow research field on aircraft duct defects and the lack of relevant research results, and considering that aircraft ducts are steel objects with small detection targets, we will conduct more in-depth research on steel object defect detection and small object detection. In terms of model improvements, Wang et al. [19] improved the YOLOv8s model by adding a small object detection layer to address the issue of small object detection loss in the YOLOv8 model, focusing on the problem of small gear defects in automobiles. Li et al. [20] proposed a novel lightweight convolutional technique called GSCov, which could be applied to lightweight models while maintaining accuracy. The algorithm based on deep learning has been widely applied to establish a reliable steel surface defect detection system, which has guiding significance for aircraft duct defect detection. The performance of deep learning models heavily relies on rich annotated data. However, the volume of labeled images in industrial datasets is often limited. To address this issue, Wang et al. [21] proposed the first few-sample defect detection framework. By using data related to the target task for pre-training the model, the proposed framework can generate well-trained networks with a small number of labeled images, and a noise regularization strategy was designed to significantly improve the robustness of the training model. Zhang et al. [22] proposed a dense non-anchored rotating-object detector (DARDet) for detecting rotating objects in aerial images to address the issue of the sensitivity of rotation detection to anchoring parameters and potential performance degradation due to boundary discontinuities. They also introduced a loss of PIoU to achieve accurate and stable regression. This method achieves a state-of-the-art performance while maintaining high efficiency on three commonly used aerial object datasets, namely DOTA, HRSC2016, and UCAS-AOD. Wang et al. [23] designed an efficient anchor-free rotating-object detector based on PP-YOLOE. Many useful techniques have been introduced to PP-YOLOE-R to improve the detection accuracy with few additional parameters and computational costs. The results showed that PP-YOLOE-Rl and PP-YOLOE-R-x achieved 78.14 and 78.28 mAP, respectively, on the DOTA 1.0 dataset, which were almost superior to all other rotating object detectors. Although the above research has made contributions to steel surface defect detection tasks and lightweight models, there is no algorithm with high robustness and accuracy in aircraft duct defect detection. Therefore, we have proposed an innovative defect detection framework based on the anchor-free model, which uses the mixed-channel attention mechanism to enhance the problem of spatial feature information extraction and greatly improve the model accuracy. In addition, we added the light enhancement module to ensure the robustness of the model in a low-illumination detection environment,

and used the space-to-depth revolution module to reduce the amount of computation while reducing the loss of information, so that our framework can be well deployed in edge computing equipment to achieve an outstanding performance and ensure that the framework is lightweight, accurate, and robust.

### 3. Proposed Method

The overall structure of LESM-YOLO is shown in Figure 1, which comprises three key modules: a low-light enhancement module, an SPDConv-based Backbone module, and an MLCA-based neck module. The original YOLOv8 model lacks the ability to identify defects in data collected under low-light conditions and tends to overlook certain details. To overcome these limitations, this study introduces the SPDConv-based backbone module [24,25] and MLCA-based [26] neck module. While reducing the number of parameters to make the model more lightweight, it also takes into account spatial feature information and performs low-light enhancement processing. This enables the efficient and precise detection of small target defects in aircraft ducts, even with unchanged inputs. In this chapter, we will take a closer look at what each module does and how it works.



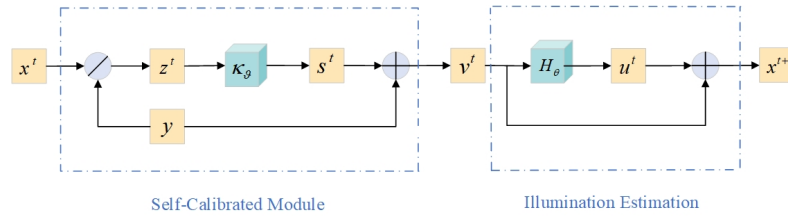
**Figure 1.** Architecture of LESM-YOLO. We added the light enhancement module before the convolution operation, changed the original Conv convolution in the backbone to SPDConv convolution, and finally added a lightweight MLCA attention mechanism in the neck to enhance the model's ability to extract spatial feature information.

#### 3.1. Low-Light Enhancement Module

The collection of data from aircraft ducts mostly takes place under low-light conditions. Due to the influence of insufficient lighting on the collected data, the quality and clarity of the images may decrease, making it difficult to distinguish defects in the dataset. At the same time, the information captured by the camera will be limited, resulting in a lack of rich details and information in the dataset, which may limit the performance and generalization ability of machine learning models in dark conditions. If low-light datasets are directly annotated, this will lead to errors or inconsistencies being flagged, affecting the quality and reliability of the training data.

To overcome the above problems, we reconstructed the YOLOv8 network and introduced a low-light enhancement module [27] into the model's backbone to optimize image quality. During the training phase, this module includes a self-calibration module and an

illumination estimation module, which facilitates weight sharing in illumination learning, causing the results of each stage to converge and thereby improving the exposure stability and significantly reducing computational noise. The structure of the low-light enhancement model is shown in Figure 2.



**Figure 2.** Low-light enhancement module framework.

In the initial stage, the images captured under raw illumination are first processed through the illumination estimation module. In the subsequent  $t - 1$  stages, the images from the previous stage are first processed through the self-calibration module before undergoing computation in the illumination estimation module.

In the self-calibrated stage, since the input of each stage comes from the output of the previous stage, the core idea is to combine the input of each stage with weak light observations for calibration to achieve convergence. In the self-calibrated stage, as shown in Figure 2,  $x^t$  represents the illumination condition at stage  $t$ ,  $y$  is the low-light observation, and  $z^t$  is the desired clear image at stage  $t$ .  $K$  denotes the shared weights and parameters, and  $v^t$  is the transformed input after passing through the self-calibration module at stage  $t$ .

Here,  $F$  represents the illumination estimation module, which includes the following equations:

$$F(x^t) = \begin{cases} u^t = H_\theta(x^t) \\ x^{t+1} = x^t + u^t \end{cases} \quad (1)$$

where  $H_\theta$  represents shared weights, which means that the same architecture  $H$  and parameters are used at each stage  $\theta$ . The illumination  $x^{t+1}$  in stage  $t + 1$  is composed of the residual light  $u^t$  and illumination  $x^t$  from the previous stage  $t$ . The structure also incorporates residual representation to ensure the stability of exposure. In the self-calibrated stage, since the input of each stage comes from the output of the previous stage, the core idea is to combine the input of each stage with weak light observations for calibration to achieve convergence.

The low-light enhancement module uses specific loss functions to maintain the color fidelity, contrast, and detail of the enhanced images. These loss functions ensure that the enhanced images have good visual quality. The loss function is shown in Equation (2):

$$L_{total} = \alpha L_f + \beta L_s \quad (2)$$

where  $L_f$  and  $L_s$  represent fidelity and smoothness losses, respectively. We adopt fidelity loss [27], represented as follows:

$$L_f = \sum_{t=1}^T \|x^t - (y + s^{t-1})\|^2 \quad (3)$$

where  $y + s^{t-1}$  is used to constrain the illumination  $x^t$ . At the same time, smoothness loss is a broad consensus in this task [28,29]. The formula is as follows:

$$L_s = \sum_{i=1}^N \sum_{j \in N(i)} w_{i,j} |x_i^t - x_j^t| \quad (4)$$

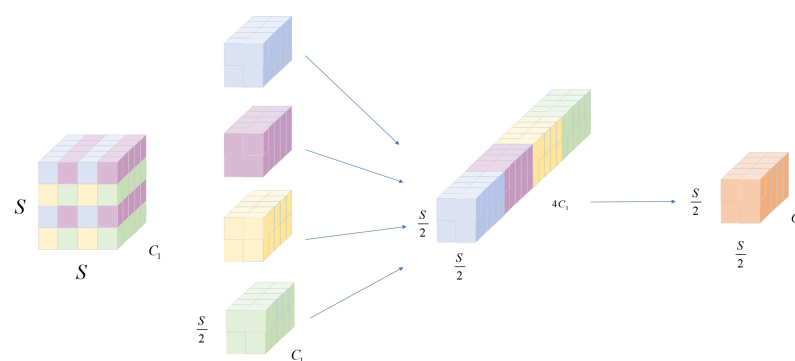
### 3.2. SPDCConv-Based Backbone

Convolutional neural networks (CNNs) have achieved significant results in various fields, such as image classification, object detection, image segmentation, and medical

image analysis. However, in difficult tasks with a low image resolution or small objects, such as when identifying small targets such as cracks, scratches, and defects in aerial ducts, there are shortcomings, such as information granularity loss and a large parameter quantity. Therefore, we introduced space-to-depth convolution (SPDConv) [24,25] to optimize the original YOLOv8 network model architecture so that the model can reduce the number of parameters while maintaining the accuracy and speed of its defect detection.

In the backbone section of YOLOv8, the original convolution operations primarily focus on local features, with the output feature map size determined by the input size, kernel size, stride, and padding. This approach may overlook global contextual information, which is crucial for the classification of defects in aircraft ducts. SPDConv is mainly used for reordering feature maps, allowing spatial information to be transformed into depth information while preserving the original details. This operation is typically undertaken to enhance the expressive capacity of feature maps, making it easier for the model to capture complex spatial relationships. It is not a process of weighted summation, but rather a reorganization process. By reorganizing the data structure of the feature map, its shape is altered to better integrate features from different levels.

LESM-YOLO modified the original YOLOv8 model by replacing the original convolution module with a new convolution module consisting of space-to-depth and one-strided convolution layers, based on the original backbone, to eliminate the convolution and pooling layers in each layer. This convolution operation divides the input tensor into blocks according to their spatial dimensions and rearranges them, increasing their depth direction while reducing their spatial direction. The space to depth convolution structure is shown in Figure 3. The specific implementation method is as follows: first, the input tensor is divided according to the specified spatial block size, and each block is rearranged into deeper tensors. This process will reduce the spatial dimensions of the input tensor and increase the depth dimensions. For example, if the size of the input tensor is  $(B, S, S, C_1)$ , where  $B$  represents the batch size,  $S$  represents the height and width, and  $C_1$  represents the number of channels, then the tensor size after spatial partitioning and rearrangement may become  $(B, \frac{S}{k}, \frac{S}{k}, C_1 \times k^2)$ , where  $k$  is the size of the spatial partitioning. Finally, a convolutional layer with a stride of 1 is used, the number of channels is set to  $C_2$ , and a stride of 1 convolution is used to preserve all feature information. Finally, the image output is  $(B, \frac{S}{k}, \frac{S}{k}, C_2)$ . Space-to-depth convolution can effectively preserve feature information with fewer parameters and lower computational costs than the original convolution. By utilizing the sparsity of spatial information and reorganizing in the depth direction to reduce computational complexity, it can accurately detect small defects and can be well applied in aircraft duct defect detection problems.



**Figure 3.** Principle diagram of space to depth convolution.

### 3.3. MLCA-Based Neck

Currently, the enhancement of object detection models mainly focuses on loss function optimization, network structure optimization, and data augmentation. The attention mechanism is a key component of network structure optimization, as it helps the model better focus on important parts of the image and suppress irrelevant elements, thereby

improving detection accuracy and efficiency. However, most channel attention mechanisms only include feature information and ignore spatial feature information, which can lead to poor model representation or a poor object detection performance. Some attention mechanisms do incorporate spatial information, but their computational and parameter requirements are too high, which often results in reduced accuracy due to channel dimension reductions during fusion. To achieve a balance between accuracy and complexity in aircraft duct defect detection and address the issues present in the aforementioned attention mechanisms, this paper introduces a neck module based on mixed local channel attention (MLCA-based neck).

This module consists of two parts: local channel attention and spatial attention. Local channel attention enhances the discriminative power of feature maps by emphasizing important channels and suppressing irrelevant ones. It calculates channel attention weights through global average pooling (GAP) and a series of fully connected layers. Spatial attention aims to highlight important regions within each feature map by aggregating spatial information through convolutional layers and generating a spatial attention map.

The MLCA-based neck module primarily consists of the following steps: First, feature maps are extracted from the backbone network, denoted as  $F \in R^{C \times H \times W}$ , where  $C$  is the number of channels, and  $H$  and  $W$  are the height and width of the feature maps, respectively. In the local channel attention mechanism part, global average pooling is applied to the feature maps to obtain channel descriptors, as shown in Equation (5). These descriptors are passed through a small feed-forward network (usually consisting of one or two fully connected layers) to compute the channel attention weights, as shown in Equation (6).

$$f_c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W F_{cij} \quad (5)$$

$$M_c = \sigma(W_1 \delta(W_0 f_c)) \quad (6)$$

Here,  $\sigma$  is the sigmoid activation function,  $\delta$  is the ReLU activation function, and  $W_0$  and  $W_1$  are learnable weight matrices.

As shown in Equation (7), the spatial attention mechanism applies convolutional layers (usually with a  $7 \times 7$  kernel) to the feature maps to capture spatial dependencies and generate spatial attention maps.

$$M_s = \sigma(\text{Conv}(F)) \quad (7)$$

where  $\text{Conv}$  represents the convolution operation and  $\sigma$  is the sigmoid activation function.

Finally, the original feature maps are combined with the computed channel and spatial attention maps to enhance the features, as expressed in Equation (8). This design allows for the simultaneous consideration of both channel and spatial information, enhancing relevant features while suppressing irrelevant ones.

$$F' = F \times M_c \times M_s \quad (8)$$

## 4. Experiments and Analysis

### 4.1. Experimental Environment

To verify the effectiveness of our proposed method, we used the Ubuntu operating system, Pytorch 2.1.0 as the deep learning framework, and YOLOv8s as the baseline network model. The specific configuration of the experimental environment is shown in Table 1.

**Table 1.** Configuration and training environment.

Environmental Parameter	Value
System environment	Ubuntu 22.04
Deep learning framework	PyTorch 2.1.0
Cuda version	12.1
GPU	RTX 4090 (24 GB)
CPU	Intel(R) Xeon(R) Platinum 8352V CPU @ 2.10 GHz
Programming language	Python 3.10

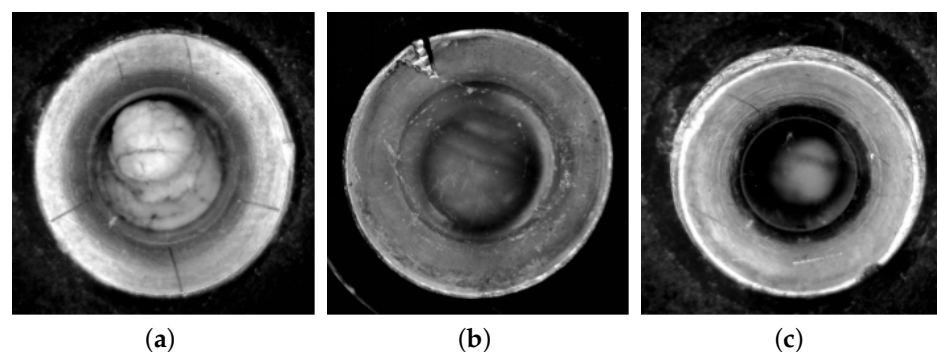
During the training phase, we adopted hyperparameter settings as shown in Table 2.

**Table 2.** Hyperparametric configuration.

Hyperparameters	Value
Learning rate	0.01
Image size	640 × 640
Momentum	0.937
Batch size	4
Epoch	150
Weight decay	0.0005

#### 4.2. Dataset and Evaluation Metrics

In this study, a specially designed instrument was used to collect data on internal defects in aircraft ducts. The images are grayscale with a resolution of 1100 × 1100, and a total of 1800 images were collected. For aircraft ducts of varying diameters, we employed different diameter-specific acquisition devices to complete the data collection. In the dataset of 1800 images, the training set contains 1250 images, the validation set contains 270 images, and the test set contains 280 images. Among these, there are 1746 cracks, 2153 scratches, and 2483 defects. Each image reflects different defect issues in the actual use of aircraft ducts. To ensure the accuracy of detecting defect locations, the roLabelImg image annotation software was used to manually annotate the defect areas. We classify aircraft duct defects into three categories: scratch, defect, and crack. The annotation file is saved in XML format and then converted to TXT format to adapt to the YOLOv8 algorithm. Figure 4 shows partial images of the dataset.



**Figure 4.** Different defect collection samples. The main defect in (a) is cracks, the main defect in (b) is defects, and the main defect in (c) is scratches. Each image may also contain multiple types of defect.

In addition, we used a grayscale distribution histogram to measure the brightness of the image. It can be seen that the introduction of a lighting enhancement module has an impact on the grayscale distribution histogram of the image, as shown in Figure 5.

In order to objectively evaluate the performance of defect detection in aircraft ducts, this study used Precision ( $P$ ), Recall ( $R$ ), mean Average Precision ( $mAP$ ) and Frames Per Second ( $FPS$ ) as evaluation indicators. Specifically,  $P$  represents the ratio of the predicted



algorithm area to the actual detection area and  $R$  represents the proportion of accurately predicted categories to the total required categories.  $mAP$  is used to evaluate the performance of multiple classifiers.  $FPS$  is used to evaluate the speed of object detection in terms of the number of images that can be processed per second. The volume of network parameters indicates that a smaller model size means lower memory usage. The core formula for evaluating indicators is as follows:

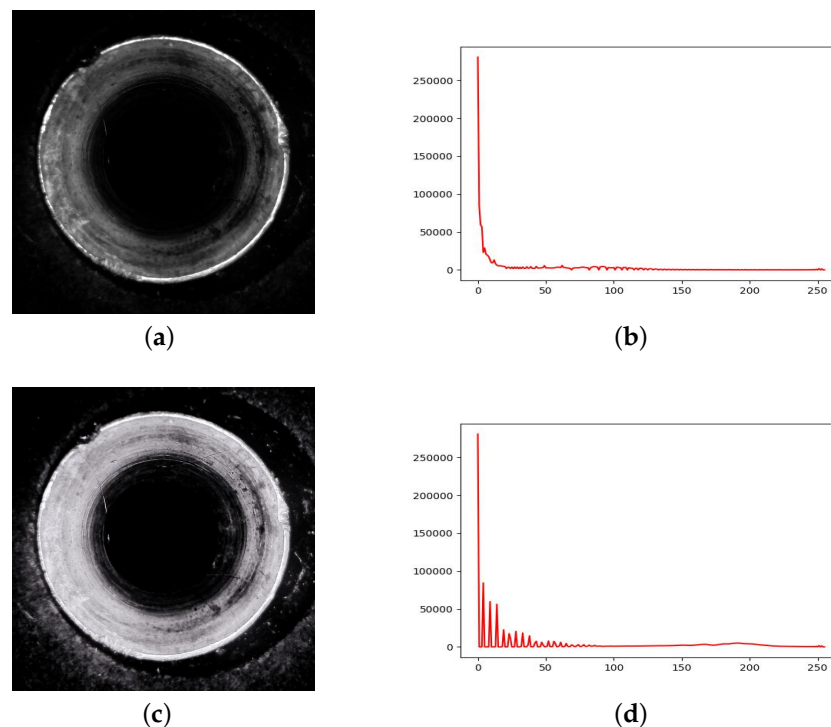
$$Precision = \frac{TP}{TP + FP} \quad (9)$$

$$Recall = \frac{TP}{TP + FN} \quad (10)$$

$$AP = \int_0^1 p(r)dr \quad (11)$$

$$mAP = \frac{1}{N} \sum AP_i \quad (12)$$

where  $N$  represents the category overview,  $TP$  represents the number of correctly identified positive samples,  $FP$  represents the number of false-positive negative samples,  $FN$  represents the number of missed positive samples, and  $TN$  represents the number of correctly identified negative samples.  $AP$  and  $mAP$  represent single-class accuracy and average accuracy, respectively.



**Figure 5.** The influence of the lighting enhancement module on the grayscale distribution histogram of images. (a) Original image. (b) Original grayscale distribution. (c) Image after lighting enhancement. (d) Grayscale distribution after lighting enhancement.

#### 4.3. Ablation Experiment

The table uses a check mark symbol ( $\checkmark$ ) to indicate the corresponding improvement strategies. Each set of experiments was conducted with the same training parameters.

Based on the comparative data in Table 3, it is verified that the use of the LE module improved the model's  $mAP$  by 2.8%. Additionally, incorporating the SPD-Conv concept to reconstruct the backbone significantly increased the model's  $FPS$  to 153.6 and the  $mAP$  value to 93.8%. The higher  $FPS$  enables the model to process images faster, achieving real-

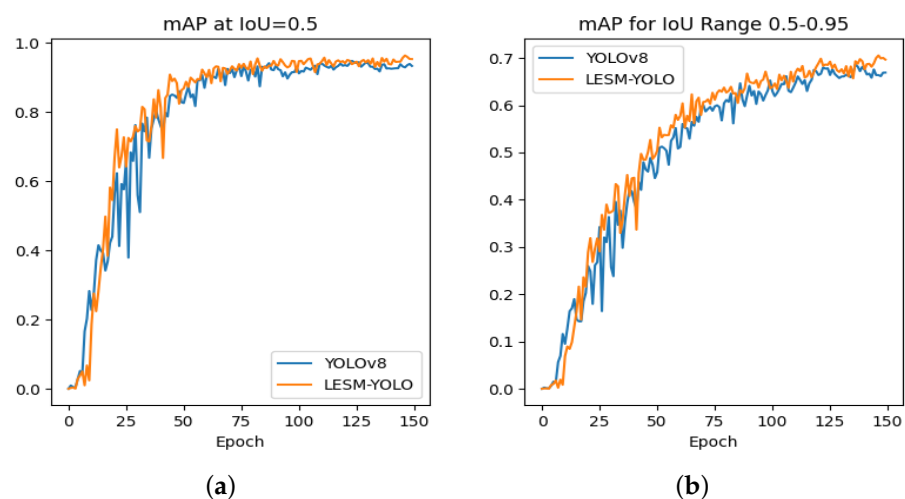
time detection and analysis, and reduces the latency between image capture and processing. The MLCA attention mechanism module, being lightweight, contributed the most to the mAP improvement, reaching 94.1%. It can be observed that the SPD-Conv reconstruction caused a slight decrease in mAP. This may be because traditional convolution methods gradually expand the receptive field by increasing the network depth, whereas SPD-Conv changes the spatial dimensions of the feature maps in a less natural manner, affecting global information integration. However, reconstructing the backbone with SPD-Conv greatly enhanced the model's FPS. As the aircraft detection task involves numerous ducts and a high workload, with stringent real-time requirements, faster defect identification can improve work efficiency and reduce downtime for maintenance. Combining the LE module, SPD-Conv, and MLCA attention mechanism to reconstruct the YOLOv8 model achieved a balance between accuracy, speed, and algorithm robustness. The results show an mAP of 96.3 and an FPS of 138.7.

**Table 3.** Ablation experiments with the modules.

LE-Module	SPD-Conv	MLCA	P	R	mAP	FPS
			87.5	85.7	89.9	140.3
✓			89.6	91.3	92.7	135.8
	✓		91.7	92.1	93.8	153.6
		✓	91.4	89.9	94.1	128.9
✓		✓	95.5	90.1	97.1	124.4
✓	✓	✓	94.8	92.8	96.3	138.7

✓ indicates the corresponding improvement strategy.

As shown in Figure 6, our model outperforms the original model in both mAP@0.5 and mAP@0.5:0.95 metrics. Initially, YOLOv8 shows a faster increase in mAP, but after 25 epochs, LESM-YOLO consistently achieves higher mAP@0.5 and mAP@0.5:0.95 metrics than the original model.



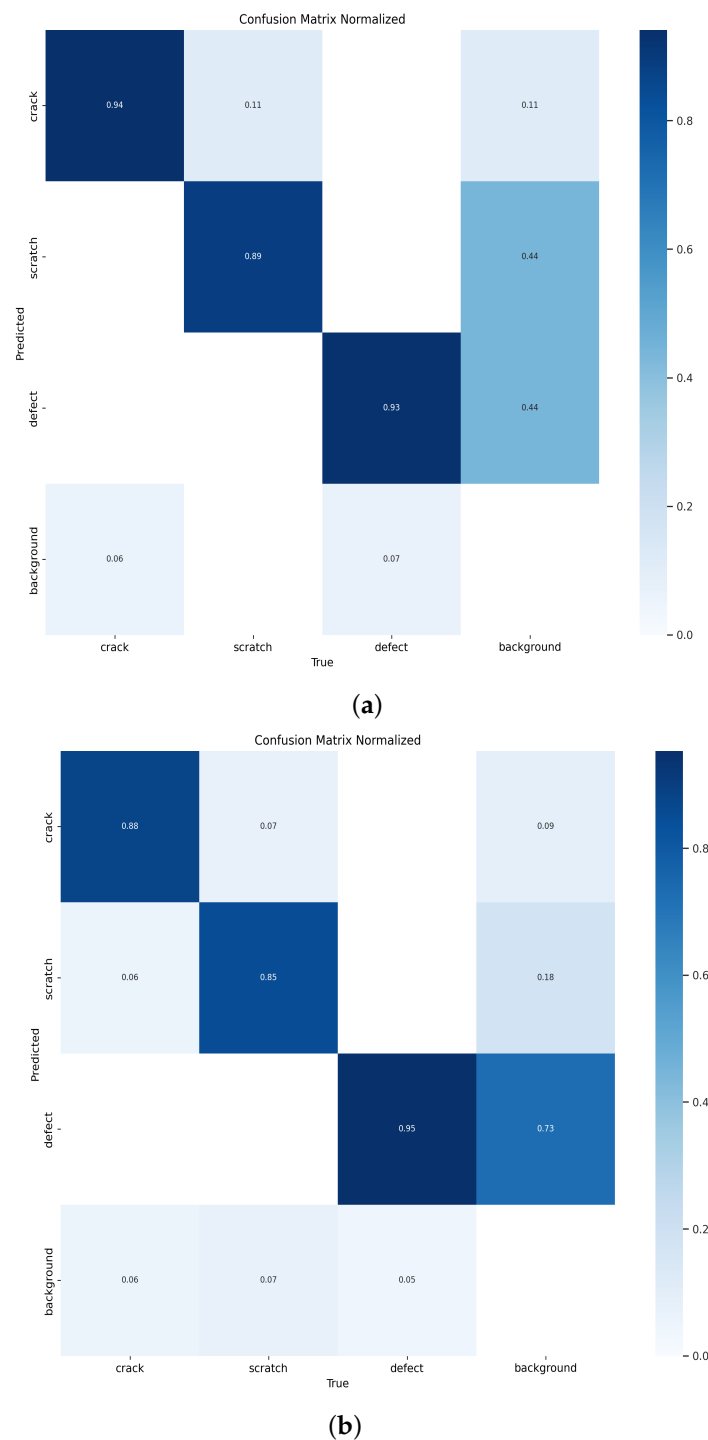
**Figure 6.** A comparison was conducted to evaluate the mAP values of the LESM-YOLO model against the original. (a) Comparison of mAP@0.5 and (b) comparison of mAP@0.5–0.95.

#### 4.4. Interpretability Experiment

In deep learning, the confusion matrix is a tool used to evaluate the performance of classification models. It compares the model's predictions with the true labels and categorizes them into four different scenarios: True Positive ( $TP$ ), True Negative ( $TN$ ), False Positive ( $FP$ ), and False Negative ( $FN$ ). Through the confusion matrix, the classification performance of the model for different classes can be obtained, providing us with a more comprehensive understanding of the model's performance. In this experiment, we chose to

compare the LESM-YOLO model with the YOLOv8 model and analyzed their confusion matrices to validate the performance of the proposed model.

From the Figure 7, it can be seen that when identifying scratch defects, the original YOLOv8 model had an accuracy of only 85%, while the LESM-YOLO model improved the accuracy by 4%. When identifying crack defects, the accuracy of the LESM-YOLO reached 94%, which is 6% higher than the original model. Therefore, the proposed LESM-YOLO significantly improves the accuracy of defect identification and exhibits a superior detection performance.



**Figure 7.** Confusion matrix of the LESM-YOLO model and YOLOv8 model. (a) Confusion matrix of LESM-YOLO. (b) Confusion matrix of YOLOv8.

#### 4.5. Comparison of Performance of Different Models

To evaluate the performance enhancement of the augmented model, this study conducted comparative experiments between the augmented model and various widely used object detection models. The selected models include two-stage anchor-based methods such as Faster R-CNN, as well as single-stage anchor-based methods such as SSD, YOLOv3, YOLOv4-tiny, YOLOv5, and YOLOv7, in addition to the Transformer-based YOLOs. Additionally, the experiments were conducted on the same dataset and under identical experimental conditions.

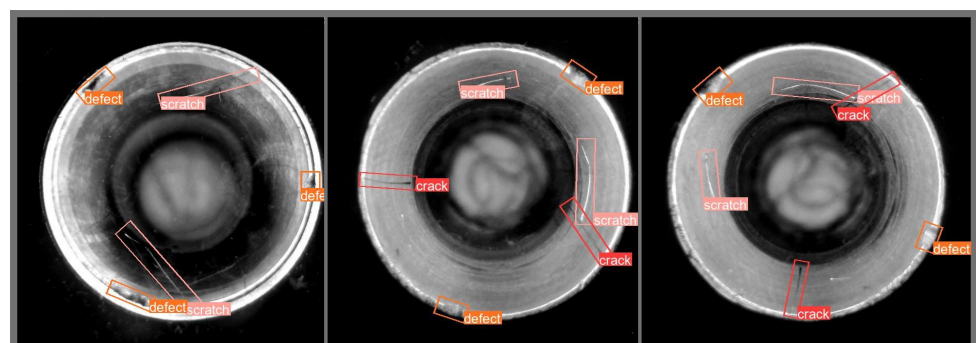
According to the results in Table 4, we found that, compared to the SSD model, our proposed LESM-YOLO model achieved a 4.67% improvement in the mAP performance metric and a 3.21 times speedup in detection. Additionally, compared to YOLOv8s and YOLOv7-tiny, with similar detection speeds, the model's mAP increased by 5.4% and 8.87% respectively, achieving very satisfactory results. Compared to two-stage algorithms represented by Faster R-CNN, our LESM-YOLO model exhibits more significant advantages in both detection speed and accuracy.

**Table 4.** Comparison of detection performance among different models.

Models	Crack AP(%)	Scratch AP(%)	Defect AP(%)	mAP50	FPS
Faster-RCNN	80.89	72.49	73.81	75.73	9.6
SSD	94.34	89.86	90.69	91.63	43.2
YOLOv3	85.15	81.06	83.21	83.14	54.1
YOLOv4-tiny	81.81	77.54	80.02	79.79	145.3
YOLOv5	88.93	84.82	86.41	86.72	98.2
YOLOv7-tiny	90.13	84.63	87.53	87.43	102.3
YOLOS-Ti	88.64	85.18	86.67	86.83	113.6
YOLOv8s	90.82	86.76	95.12	90.90	140.3
Our Model	97.71	94.45	96.74	96.30	138.7

Similarly, we conducted a performance comparison with the YOLOs (You Only Look at One Sequence) model [30]. YOLOs leverages the self-attention mechanism of the transformer to capture both global and local features in images. Our results demonstrate that our model outperforms YOLOs-Ti, achieving a 9% higher mAP50 score, which is a significant improvement. These findings indicate that the LESM-YOLO model exhibits superior performance and stability, with outstanding accuracy in detecting three types of defects.

Finally, our LESM-YOLO model was employed to detect the aircraft duct defect dataset. The detection results are shown in Figure 8. We utilized rotated boxes of different colors to represent the detection results for various types of defects. Specifically, red boxes indicate cracks, orange boxes indicate defects, and pink boxes indicate scratches.



**Figure 8.** Actual detection results of the LESM-YOLO model.

## 5. Conclusions

Efficient and accurate detection is crucial for ensuring smooth aircraft operation. This paper proposes an aircraft duct defect detection model based on YOLOv8. In this approach, we incorporated a low-light enhancement module to effectively handle defect detection images captured under low-light conditions, thereby enhancing the accuracy and robustness of the model. Subsequently, we replaced the original convolutional layers with space-to-depth convolutions in the backbone section to reduce the parameter count of the model, making it suitable for deployment on edge detection devices while maintaining the defect detection accuracy. Finally, we improved the neck module of the original model by designing an MLCA-based neck, further enhancing the detection accuracy of the model. Through multiple experiments, we validated that our model can effectively complete aircraft duct defect detection tasks under low-light conditions with high quality.

**Author Contributions:** Conceptualization: Y.Y. and R.W.; Methodology: R.W. and Y.Y.; Data Curation: Y.W., Z.L. and Q.L.; Original Draft: R.W.; Review and Editing: Z.L. and Y.C.; Software: R.W. and Q.L.; Validation: Q.L., Y.C. and Y.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data that support the findings of this study are available from the corresponding author.

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

## References

1. Chen, L.; Zou, L.; Fan, C.; Liu, Y. Feature weighting network for aircraft engine defect detection. *Int. J. Wavelets Multiresolut. Inf. Process.* **2020**, *18*, 2050012. [[CrossRef](#)]
2. Bulnes, F.G.; Usamentiaga, R.; Garcia, D.F.; Molleda, J. An efficient method for defect detection during the manufacturing of web materials. *J. Intell. Manuf.* **2016**, *27*, 431–445. [[CrossRef](#)]
3. Upadhyay, A.; Li, J.; King, S.; Addepalli, S. A Deep-Learning-Based Approach for Aircraft Engine Defect Detection. *Machines* **2023**, *11*, 192. [[CrossRef](#)]
4. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Switzerland, 2018; pp. 3–19. [[CrossRef](#)]
5. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the Computer Vision—ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; pp. 21–37.
6. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the Computer Vision & Pattern Recognition, Las Vegas, NV, USA, 21–30 June 2016. [[CrossRef](#)]
7. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [[CrossRef](#)]
8. 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](#)]
9. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141. [[CrossRef](#)]
10. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 3141–3149. [[CrossRef](#)]
11. Park, J.; Woo, S.; Lee, J.Y.; Kweon, I.S. BAM: Bottleneck Attention Module. *arXiv* **2018**, arXiv:1807.06514. <https://doi.org/10.48550/arXiv.1807.06514>.
12. Mahajan, D.; Girshick, R.; Ramanathan, V.; He, K.; Paluri, M.; Li, Y.; Bharambe, A.; van der Maaten, L. Exploring the Limits of Weakly Supervised Pretraining. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Switzerland, 2018; pp. 185–201. [[CrossRef](#)]

13. Wang, W.; Xie, E.; Li, X.; Fan, D.P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 548–558. [[CrossRef](#)]
14. Yao, Y.; Ren, J.; Xie, X.; Liu, W.; Liu, Y.J.; Wang, J. Attention-Aware Multi-Stroke Style Transfer. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 1467–1475. [[CrossRef](#)]
15. Zamir, S.W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.H.; Shao, L. Learning Enriched Features for Real Image Restoration and Enhancement. In Proceedings of the Computer Vision—ECCV 2020, Glasgow, UK, 23–28 August 2020; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M., Eds.; Springer: Cham, Switzerland, 2020; pp. 492–511. [[CrossRef](#)]
16. Wang, T.C.; Mallya, A.; Liu, M.Y. One-Shot Free-View Neural Talking-Head Synthesis for Video Conferencing. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 10034–10044. [[CrossRef](#)]
17. Su, Y.; Liu, X.; Guo, X. A Target Tracking Algorithm Based on Channel Attention and Spatial Attention. In Proceedings of the 2023 3rd International Symposium on Artificial Intelligence and Intelligent Manufacturing (AIIM), Chengdu, China, 27–29 October 2023; pp. 57–61. [[CrossRef](#)]
18. Han, Y.; Huang, G.; Song, S.; Yang, L.; Wang, H.; Wang, Y. Dynamic Neural Networks: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 7436–7456. [[CrossRef](#)] [[PubMed](#)]
19. Wang, H.; Yang, G.; Hao, X.; Geng, L. Automotive Gear Defect Detection Method based on Yolov8 Algorithm. In Proceedings of the 2023 Asia Symposium on Image Processing (ASIP), Tianjin, China, 15–17 June 2023; pp. 19–23. [[CrossRef](#)]
20. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv* **2022**, arXiv:2206.02424.
21. Wang, H.; Li, Z.; Wang, H. Few-Shot Steel Surface Defect Detection. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 5003912. [[CrossRef](#)]
22. Zhang, F.; Wang, X.; Zhou, S.; Wang, Y. DARDet: A Dense Anchor-Free Rotated Object Detector in Aerial Images. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 8024305. [[CrossRef](#)]
23. Wang, X.; Wang, G.; Dang, Q.; Liu, Y.; Hu, X.; Yu, D. PP-YOLOE-R: An efficient anchor-free rotated object detector. *arXiv* **2022**, arXiv:2211.02386.
24. Sunkara, R.; Luo, T. No More Strided Convolutions or Pooling: A New CNN Building Block for Low-Resolution Images and Small Objects. In Proceedings of the Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Grenoble, France, 19–23 September 2022; Springer: Berlin/Heidelberg, Germany, 2023; pp. 443–459. [[CrossRef](#)]
25. Yang, Z.; Wu, Q.; Zhang, F.; Zhang, X.; Chen, X.; Gao, Y. A New Semantic Segmentation Method for Remote Sensing Images Integrating Coordinate Attention and SPD-Conv. *Symmetry* **2023**, *15*, 1037. [[CrossRef](#)]
26. Wan, D.; Lu, R.; Shen, S.; Xu, T.; Lang, X.; Ren, Z. Mixed local channel attention for object detection. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106442. [[CrossRef](#)]
27. Ma, L.; Ma, T.; Liu, R.; Fan, X.; Luo, Z. Toward Fast, Flexible, and Robust Low-Light Image Enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 5637–5646.
28. Guo, C.; Li, C.; Guo, J.; Loy, C.C.; Hou, J.; Kwong, S.; Cong, R. Zero-Reference Deep Curve Estimation for Low-Light Image Enhancement. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1777–1786. [[CrossRef](#)]
29. Zhang, Y.; Guo, X.; Ma, J.; Liu, W.; Zhang, J. Beyond Brightening Low-light Images. *Int. J. Comput. Vis.* **2021**, *129*, 1013–1037. [[CrossRef](#)]
30. Fang, Y.; Liao, B.; Wang, X.; Fang, J.; Qi, J.; Wu, R.; Niu, J.; Liu, W. You only look at one sequence: Rethinking transformer in vision through object detection. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 26183–26197.

**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.