agriculture

MDPI

Article

Fine-Grained Detection Model Based on Attention Mechanism and Multi-Scale Feature Fusion for Cocoon Sorting

Han Zheng [1,†], Xueqiang Guo [2,3,†], Yuejia Ma [2], Xiaoxi Zeng [2], Jun Chen [1] and Taohong Zhang [2,3,*]

[1] Key Laboratory of AI and Information Processing, Education Department of Guangxi Zhuang Autonomous Region, Hechi University, Hechi 546300, China; 05057@hcnu.edu.cn (H.Z.); 019039@hcnu.edu.cn (J.C.)
[2] Department of Computer, School of Computer and Communication Engineering, University of Science and Technology Beijing (USTB), Beijing 100083, China; m202110626@xs.ustb.edu.cn (X.G.); u202341643@xs.ustb.edu.cn (Y.M.); u202141928@xs.ustb.edu.cn (X.Z.)
[3] Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing 100083, China
* Correspondence: zth_ustb@163.com
† These authors contributed equally to this work.

Abstract: Sorting unreelable inferior cocoons during the reeling process is essential for obtaining high-quality silk products. At present, silk reeling enterprises mainly rely on manual sorting, which is inefficient and labor-intensive. Automated sorting based on machine vision and sorting robots is a promising alternative. However, the accuracy and computational complexity of object detection are challenges for the practical application of automatic sorting, especially for small stains of inferior cocoons in images of densely distributed cocoons. To deal with this problem, an efficient fine-grained object detection network based on attention mechanism and multi-scale feature fusion, called AMMF-Net, is proposed for inferior silkworm cocoon recognition. In this model, fine-grained object features are key considerations to improve the detection accuracy. To efficiently extract fine-grained features of silkworm cocoon images, we designed an efficient hybrid feature extraction network (HFE-Net) that combines depth-wise separable convolution and Transformer as the backbone. It captures local and global information to extract fine-grained features of inferior silkworm cocoon images, improving the representation ability of the network. An efficient multi-scale feature fusion module (EMFF) is proposed as the neck of the object detection structure. It improves the typical down-sampling method of multi-scale feature fusion to avoid the loss of key information and achieve better performance. Our method is trained and evaluated on a dataset collected from multiple inferior cocoons. Extensive experiments validated the effectiveness and generalization performance of the HFE-Net network and the EMFF module, and the proposed AMMF-Net achieved the best detection results compared to other popular deep neural networks.

Keywords: attention mechanism; multi-scale feature fusion; object detection; inferior cocoon recognition; sericulture

Citation: Zheng, H.; Guo, X.; Ma, Y.; Zeng, X.; Chen, J.; Zhang, T. Fine-Grained Detection Model Based on Attention Mechanism and Multi-Scale Feature Fusion for Cocoon Sorting. *Agriculture* **2024**, *14*, 700. https://doi.org/10.3390/agriculture14050700

Academic Editor: Jiehao Li

Received: 18 March 2024
Revised: 24 April 2024
Accepted: 25 April 2024
Published: 29 April 2024

1. Introduction

Silk garments are popular worldwide and have become a symbol of high-end fashion due to their lightness, softness, elegance, and comfortable texture. The production of silk consists of the following stages: cocoon breeding and collection, cocoon harvesting, reeling, silk weaving, printing and dyeing, and manufacturing of the finished product. Each step in the above chain requires specialized skills and experience to ensure that final silk products, such as raw silk, silk, garments, apparel, and home textiles, are of high quality and aesthetically pleasing.

With the booming development of artificial intelligence technologies, the mulberry sericulture industry has witnessed a transformation in the process of production and management with intelligence and automation. Such changes have already yielded significant

achievements in several key areas of the industry. Researchers have employed deep learning techniques to enhance traditional sericulture practices. By collecting and analyzing data, they enable real-time monitoring of the growth environment and health status of silkworms [1–3], facilitating intelligent breeding approaches. Furthermore, image processing algorithms are utilized to automate the counting of silkworm eggs [4], addressing issues such as egg damage, labor-intensive processes, and inefficiency associated with manual counting. Additionally, machine vision algorithms are applied for the gender classification of silkworm cocoons [5–8]. Since male cocoons yield finer silk than female cocoons, this classification helps to minimize the mixing of varying silk qualities, thereby enhancing the production of superior silk within the mulberry sericulture industry.

In reeling enterprises, production techniques and processes need to be strictly controlled to obtain higher quality raw silk. The basic and key part is to select high-quality cocoons. At present, reeling enterprises mainly rely on manual sorting to pick cocoons that cannot be reeled. However, this sorting method is inefficient and requires a high intensity of manual labor. Additionally, the sorting quality of this method relies on the subjective consciousness of workers, and the poor environment in which cocoon sorting takes place will expose workers to irreversible health hazards. Automatic sorting is urgent and a good alternative method in reeling enterprises. The identification of inferior cocoons is the first step in automatic sorting. The present related research has developed from using image processing algorithms to deep learning AI models to identify and classify cocoon characteristics. For example, Vasta et al. [9] used multiple cameras and image processing algorithms to identify the shapes, sizes, and external stains of silkworm cocoons, respectively, improving existing sorting solutions through multi-step methods and machine learning algorithms. Yang et al. [10] proposed a machine vision-based cocoon quality detection algorithm, which realizes the identification and detection of single cocoons, douppion cocoons, yellow-spotted cocoons, and thin-skinned cocoons. Li et al. [11] proposed an improved YOLOv3 group cocoon species detection model to achieve the rapid and accurate identification of some group cocoon species. However, the above studies could only recognize cocoons with large defects and obvious features, and they failed to deal with cocoons with small differences. As depicted in Figure 1, the actual sorting environment is characterized by a high density of cocoons, which poses a challenge for efficient and accurate fine-grained identification using conventional deep learning methods and image processing algorithms. Specifically, cocoons labeled with rectangular and elliptical boxes are categorized as having inferior quality. While the cocoons enclosed by elliptical boxes can be readily identified due to distinct visible defects, those marked by rectangular boxes exhibit less apparent defects. This makes it difficult for standard object detection algorithms to accurately identify and classify them. Consequently, there is a need to develop specialized fine-grained detection algorithms tailored to discern the subtle differences in the inferior cocoons labeled with rectangular boxes.

Feature extraction is a critical component in the field of deep learning as it dictates the model's ability to comprehend the input image data. Convolutional Neural Networks (CNNs) and Transformers are recognized as two potent approaches for feature extraction, each with unique strengths that contribute to the advancement of various applications within the domain. The basic structure of a CNN consists of the following four main parts: convolutional layers, activation functions, pooling layers, and fully connected layers. Among them, convolutional layers utilize convolutional kernels to slide over the input feature map and perform element-wise multiplication and accumulation, thus capturing the local features of the input data. Pooling layers are used to achieve a reduction in the spatial dimensions of the feature map to retain the most important information. Fully connected layers are used to flatten the outputs of the convolutional and pooling layers and output the final prediction. Activation functions are used to introduce nonlinearities to the network, thus facilitating the model to learn complex patterns. CNNs [12–14] excel at capturing local patterns through convolutional and pooling layers, which not only facilitate the extraction of textural and shape features but also reduce the computational complexity, thereby

enhancing the model's generalization. Notably, CNNs have been widely implemented in various industries and achieved great progress due to their strong local perception and trainable efficiency. However, considering that the actual cocoon sorting environment is highly dense, the differences between cocoon categories are minor, and the defects in inferior cocoons occupy fewer pixels, directly applying CNNs to extract features usually results in poor model performance. On the other hand, Transformer [15] is based on the self-attention mechanism and adjusts feature representation according to the correlation of different positions in the input sequence. Its main advantage is that it allows the model to simultaneously consider all elements in the sequence when processing the sequence rather than relying only on local or adjacent information, thus facilitating the model to globally model the input image. ViT [16] and its successors [17–21] have demonstrated the potential to tackle vision tasks by processing image patches through Transformers, yet they often require extensive datasets and sophisticated training strategies to achieve competitive performance. Despite their advancements, Transformers face challenges in localized feature extraction and exhibit a quadratic increase in computational complexity with higher image resolutions, which can be impractical for certain applications.

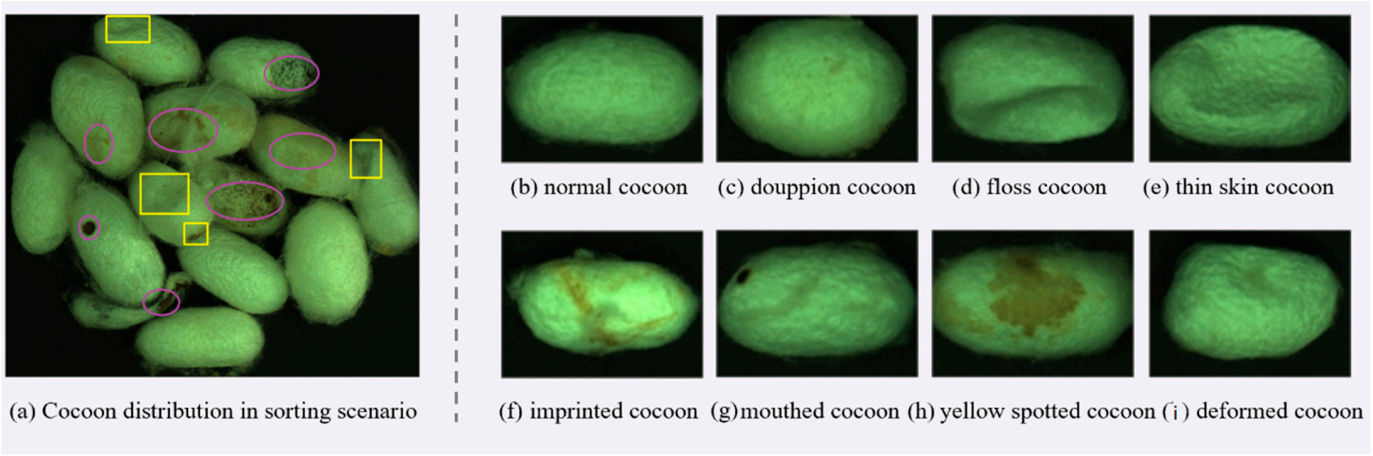

Figure 1. Example diagram of cocoon distribution in a sorting scenario and images of different categories of mulberry cocoons.

For the fine-grained recognition of silkworm cocoons, it is crucial to efficiently extract fine-grained features such as the shapes and spot defects of inferior cocoons. Meanwhile, in scenarios where cocoons are densely packed, potentially occluded, or interconnected, the intricate silk's intertwining amplifies the complexity. Therefore, it is equally important to obtain global information, which can help the model better understand the complex correlations between cocoons and provide a more comprehensive feature representation, thus improving the accuracy of object detection. To this end, we propose a hybrid feature extraction network (HFE-Net) based on a CNN and Transformer to combine the advantages of the above two network structures to fully extract the fine-grained features of silkworm cocoons. Additionally, to better fuse the multi-scale features obtained from the feature extraction network HFE-Net, we construct an efficient multi-scale feature fusion module, EMFF, which optimizes the downsampling method of typical multi-scale fusion modules and alleviates the problem of the loss of key detail information in the downsampling process. Based on the proposed HFE-Net and EMFF, and combined with object detection head, we propose an efficient object detector named AMMF-Net for fine-grained cocoon detection.

In summary, the main contributions of this paper are as follows:

(1) A novel hybrid network, HFE-Net, combining the advantages of both CNN and Transformer is proposed for efficiently extracting fine-grained cocoon image features.

(2) To address the problem of key information loss caused by the downsampling of existing multi-scale fusion modules, an efficient multi-scale feature fusion module, EMFF, is designed to effectively improve the fine-grained detection accuracy.

(3) Based on the proposed efficient fine-grained feature extraction network, HFE-Net, and the optimized multi-scale feature fusion module, EMFF, the object detection algorithm (AMMF-Net) for cocoon sorting is constructed. Extensive experiments demonstrate that the constructed object detection algorithm AMMF-Net achieves the best speed and accuracy trade-off.

2. Materials and Methods

2.1. Dataset

This work utilized three different datasets for training and evaluating our constructed innovative feature extraction network HFE-Net as well as the network AMMF-Net for fine-grained cocoon object detection.

Firstly, we selected ImageNet-100 [22] as the dataset for training the backbone network HFE-Net, which is a subset of the ImageNet-1K dataset in the 2012 ImageNet Large-Scale Visual Recognition Challenge. ImageNet-100 contains a total of 100 image categories, with each category containing about 1300 images for training and 50 images for validation, and its rich image samples and diverse category information help to improve the generalization ability and representation learning of our feature extraction network.

Secondly, in order to validate the performance of the constructed object detection network AMMF-Net, we chose the VOC07 + 12 dataset [23], which is a commonly used public object detection dataset containing a total of 21,503 images in 20 object categories with detailed annotation information. We utilized this dataset as a benchmark dataset to evaluate the performance and effectiveness of our approach in the object detection task. Additionally, the training and testing sets were divided in a manner in which the training and validation sets of VOC2007 and VOC2012 (with 16,551 images in total) were used as the training sets for the model, and the testing set of VOC2007 (containing 4952 images) was used as the testing set for the model.

Finally, for the fine-grained detection task of silkworm cocoons, we constructed a dataset focusing on mulberry silkworm cocoons, which contains multiple subcategories of mulberry silkworm cocoons. As shown in Figure 2a, we utilized an industrial camera setup to capture images of silkworm cocoons. Specifically, during the image acquisition process, we simulated the actual distribution of cocoons on a conveyor belt, such as the dense distribution of silkworm cocoons and occlusion. This approach allowed us to construct a dataset that closely resembles real industrial sorting scenarios. One of the captured silkworm cocoon images is displayed in Figure 2b, which illustrates the subtlety of defect features in images due to angles, as might be encountered in actual sorting environments. Then, the captured images were screened to eliminate blurred and redundant images, resulting in 2386 images. To optimize the training process of the model and improve the generalization capability of the model, we obtained a total of 14,316 images for the training of the object detection model by performing data augmentation operations on the cocoon images by flipping them horizontally and vertically and rotating them by 30 degrees, 60 degrees, 90 degrees, and 120 degrees. And the division ratio of the training set and testing set was 9:1.

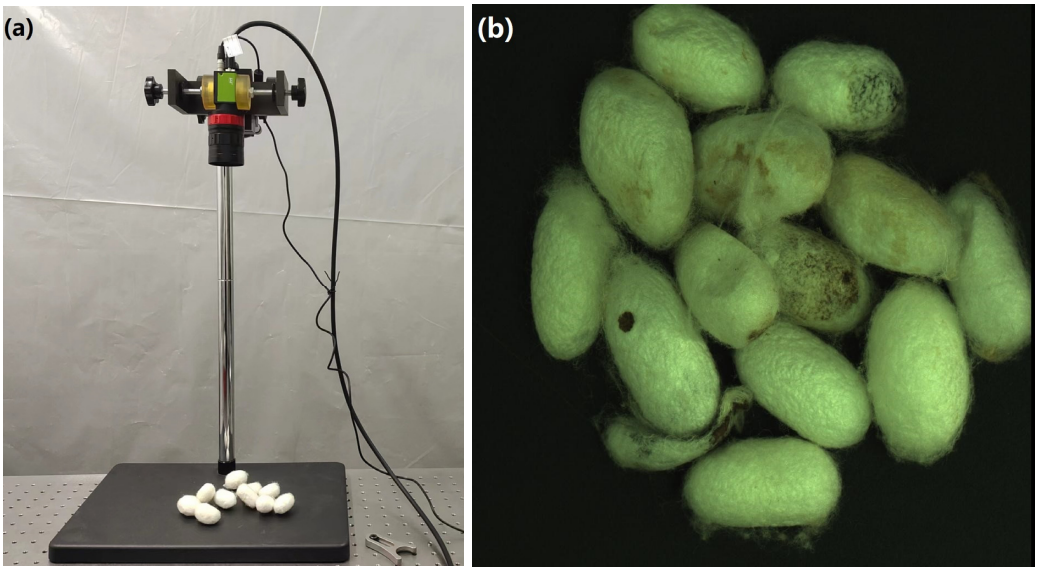

Figure 2. Image acquisition and result example. (**a**) Image capturing device; (**b**) A captured image sample.

2.2. Overview Architecture

As shown in Figure 1, the types of mulberry silkworm cocoons mainly include normal cocoons, douppion cocoons (cocoons with two or more pupae inside), floss cocoons (cocoons with loose cocoon layers and unclear wrinkles), thin skinned cocoons (cocoons with thin, inelastic layers), imprinted cocoons (cocoons with severe imprints on the surfaces of the cocoon layers), mouthed cocoons (cocoons with a hole in the cocoon layer, including moth mouth, rat mouth, maggot mouth, insect mouth, etc.), yellow spotted cocoons (cocoons with severe yellow spots on the cocoon layers), and deformed cocoons (cocoons with irregular shapes). Among them, the douppion cocoon and the normal cocoon are good cocoons that can be reeled normally, while the rest are inferior cocoons that cannot be reeled or are difficult to reel. These unreelable inferior mulberry cocoons need to be selected to improve the overall quality of raw silk. As we can see in Figure 1, the cocoon categories have a similar appearance to each other, and the spots in inferior cocoons occupy a small number of pixels, resulting in insufficient feature information to distinguish cocoon categories. In addition, the cocoons in the actual cocoon sorting environment are very dense. Therefore, it is difficult to process the above image data well using existing object detection algorithms. A detection algorithm with a fast speed and strong feature extraction capability is needed to achieve the fine-grained identification of silkworm cocoons.

This section describes our proposed method for fine-grained cocoon recognition, whose overall flowchart is shown in Figure 3. We first introduce the proposed efficient hybrid feature extraction network (HFE-Net), followed by the efficient multi-scale feature fusion module (EMFF), and then the overall fine-grained cocoon detection network based on attention mechanism and multi-scale feature fusion (AMMF-Net).

2.3. Hybrid Feature Extraction Network (HFE-Net)

Inspired by CNN and Transformer network structures, we propose a novel hybrid feature extraction network (HFE-Net) as shown in Figure 4. The proposed model combines the advantages of a CNN's robust local modeling capability and Transformer's strong global modeling capability, aiming to achieve excellent performance in fine-grained cocoon feature extraction. The network structure adopts a hierarchical structural design, where the input image resolution gradually decreases as the network stage deepens, while the semantic information is gradually enriched. Thus, the model has a multi-scale receptive field to facilitate the subsequent detection of cocoons at different scales using the extracted features.

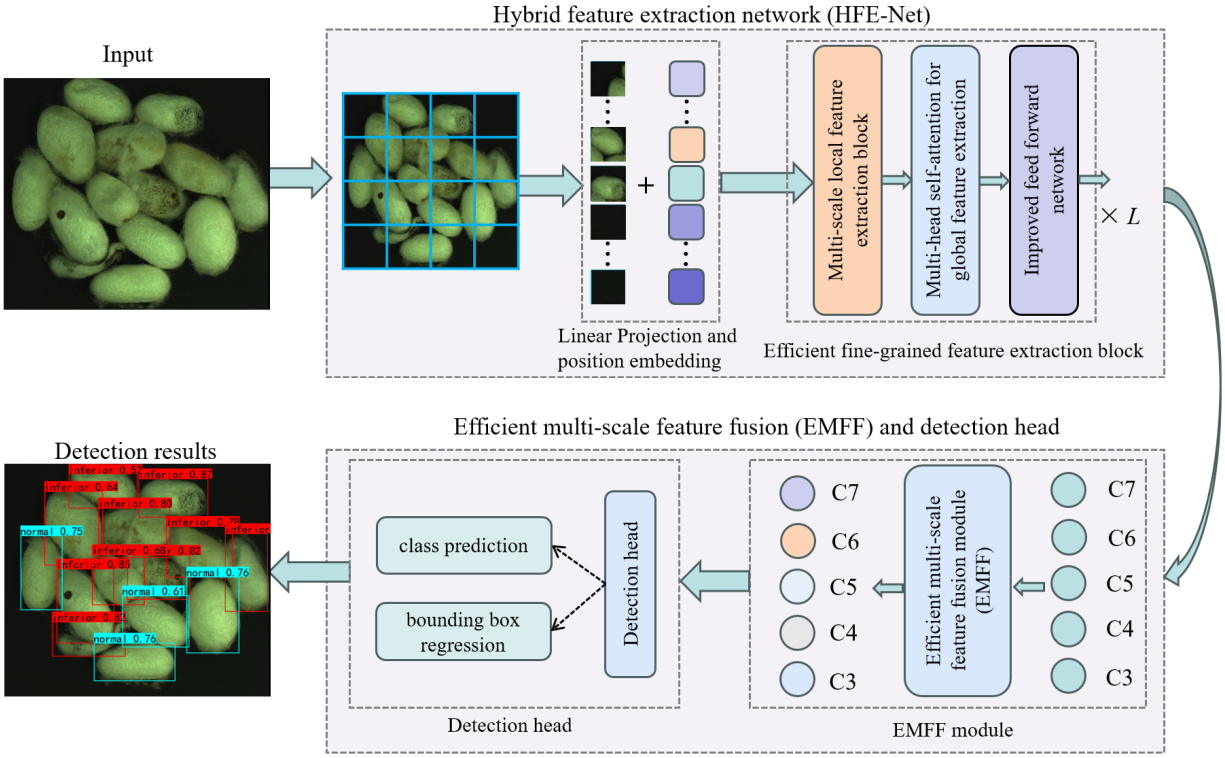

Figure 3. Proposed AMMF-Net for fine-grained cocoon detection.

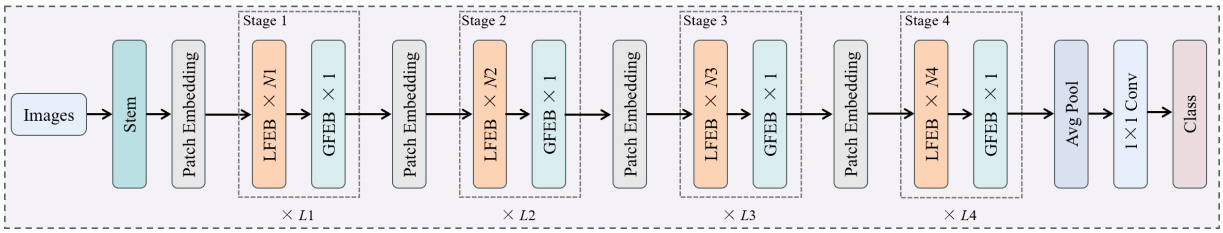

Figure 4. The overall architecture of the proposed hybrid feature extraction network (HFE-Net).

As we can see in Figure 4, the proposed HFE-Net consists of stem [24], a local feature extraction block (LFEB), and a global feature extraction block (GFEB). Among them, stem consists of a number of stacked 3 × 3 convolutions with a stride of 2 or 1, which are used to reduce the input image resolution and extract local information; LFEB is used for shallow feature extraction, which is more concerned with extracting texture detail features in local regions of the image; a GFEB is used for deep feature extraction, which is mainly used to extract the global information of the image. HFE-Net efficiently utilizes the advantages of both a CNN and Transformer, aiming to make full use of local and global information and maintain computational efficiency. Li et al. [25] pointed out that existing typical hybrid strategies (using convolutional blocks in the shallow stages and Tranformer blocks in the deep stages of the network) perform sub-optimally for dense prediction tasks (e.g., segmentation, detection, etc.). And inspired by their observation, the hybrid strategy adopted by our proposed feature extraction network involves utilizing both convolutional and Tranformer blocks at each stage of the network, which is a $(N + 1)*L$ hybrid paradigm, and it can be seen in detail in Figure 4. We will describe the LFEB and GFEB in detail below.

2.3.1. Local Feature Extraction Block (LFEB)

For fine-grained cocoon recognition, multi-scale features are crucial to correctly distinguish subtle differences between cocoons. Traditional feature extraction networks typically employ a hierarchical structure design to effectively process and recognize multi-scale fea-

tures within images. In the shallower layers of the network, feature maps maintain a higher resolution, enabling the network to capture local features such as edges and textures in the image. As the network deepens, the resolution of the feature maps gradually decreases, while the network becomes capable of extracting more advanced semantic information. Although these networks possess the capability of extracting multi-scale features, their multi-scale nature is reflected across different levels from shallow to deep areas within the network, and they do not have the capability of extracting multi-scale features within the same level. To overcome the above limitation, we propose an efficient multi-scale local feature extraction block (LFEB), as shown in Figure 5, which utilizes convolutional kernels with sizes of 1, 3, 5, and 7 to achieve feature extraction at four scales, thus enhancing the multi-scale feature extraction capability of the network and facilitating the model to extract features at a fine-grained level. Additionally, for computational reduction, we use stacked 3×3 convolutions of different numbers to achieve the equivalent performance of 5×5 and 7×7 convolutions. And the proposed LFEB can be formulated as follows:

$$\{\boldsymbol{X}_i\}_{i=1}^{4} = \text{Split}(f_{1\times1}(\boldsymbol{X})) \tag{1}$$

$$\boldsymbol{Y}_1 = \text{ReLU}(\text{BN}(f_{1\times1}(\boldsymbol{X}_1))) \tag{2}$$

$$\boldsymbol{Y}_2 = \text{ReLU}(\text{BN}(\text{DWConv}(\boldsymbol{X}_2))) \tag{3}$$

$$\boldsymbol{Y}_3 = \text{ReLU}(\text{BN}(\text{DWConv}(\boldsymbol{X}_3 + \boldsymbol{Y}_2))) \tag{4}$$

$$\boldsymbol{Y}_4 = \text{ReLU}(\text{BN}(\text{DWConv}(\boldsymbol{X}_4 + \boldsymbol{Y}_3))) \tag{5}$$

$$\boldsymbol{Y} = \text{IFFN}(f_{1\times1}(\text{Concat}(\boldsymbol{Y}_1, \boldsymbol{Y}_2, \boldsymbol{Y}_3, \boldsymbol{Y}_4)) + \boldsymbol{X}) \tag{6}$$

where $\boldsymbol{X} \in R^{H\times W\times d}$, H, and W are the input image resolutions of the current stage, respectively, and d represents the channel dimension of the features. $f_{1\times1}$ represents point convolution, Split denotes the operation of splitting the feature map $\boldsymbol{X}$ along the channel dimension, and DWConv represents the depth–width convolution. $\boldsymbol{Y}_i$ denotes the feature map produced by nonlinear activation function (ReLU [26]), batch normalization, and the convolution of different receptive field sizes. And we will introduce IFFN in detail in Section 2.3.2. The output feature map of the final fused multi-scale features is obtained by Concat operation, 1×1 convolution, and shortcut connection. Therefore, the main purpose of Equation (1) is to split the input feature map $\boldsymbol{X}$ into four sub-feature maps along the channel dimension. Equations (2)–(5) correspond to the extraction of features at different scales for each sub-feature map, respectively. Equation (6) indicates the fusion of features at four different scales and the utilization of the proposed IFFN to further enhance the expression capability of the features.

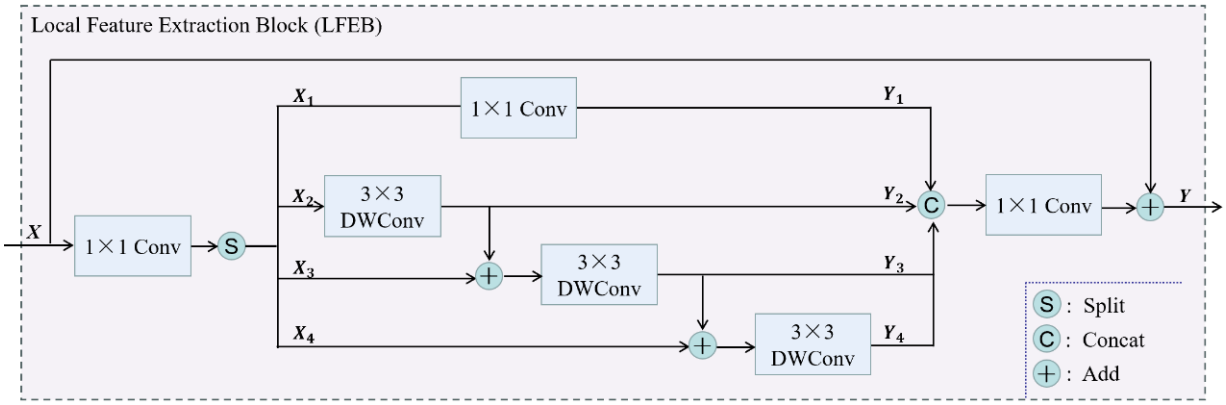

Figure 5. LFEB schematic diagram.

2.3.2. Global Feature Extraction Block (GFEB)

For fine-grained cocoon recognition, there are some defective features that are not apparent, making it hard to accurately recognize the category corresponding to the cocoon. Based on the above observation, we constructed the global feature extraction block (GFEB) based on a vision Transformer. The GFEB utilizes the attention mechanism to help the network propagate and utilize the information more efficiently and to prevent the loss of key information from occurring during the deep propagation of the network, thus improving the network's representational performance. Additionally, since the computational complexity of attention increases quadratically with the number of tokens, we apply an average pooling operation before performing the multi-head self-attention calculation to reduce the spatial dimensionality, thus reducing the number of key value pairs, which, in turn, reduces the computational cost and improves the efficiency. A schematic structural diagram of a GFEB is shown in Figure 6, and its calculation formula can be expressed as follows:

$$\boldsymbol{Y} = \mathrm{IFFN}(\mathrm{LayerNorm}(\boldsymbol{X})) + \boldsymbol{X} \tag{7}$$

$$\boldsymbol{X} = \mathrm{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \mathrm{Concat}(\mathrm{h}_1, \mathrm{h}_2 \ldots, \mathrm{h}_H) \tag{8}$$

$$\mathrm{h}_i = \mathrm{Attention}(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) \tag{9}$$

$$\mathrm{Attention}(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) = \mathrm{softmax}\left[\frac{\boldsymbol{Q}_i(\boldsymbol{K}_i)^T}{\sqrt{\frac{C}{H}}}\right]\boldsymbol{V}_i \tag{10}$$

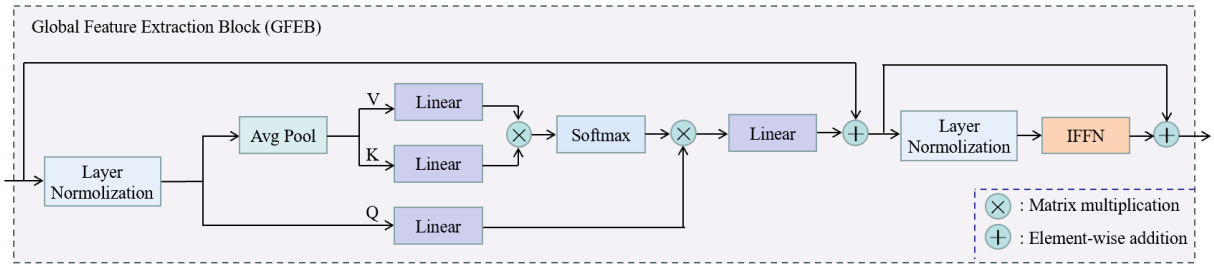

Figure 6. A schematic diagram of a GFEB.

The main purpose of Equation (7) is to apply layer normalization to the input feature map $\boldsymbol{X}$ and then pass it into the IFFN module for further feature extraction, where $\boldsymbol{Y}$ represents the output feature map, and IFFN represents the improved feed forward network proposed in this paper. Equations (8)–(10) represent the calculation operation of the attention mechanism, where h_i denotes the i-th head of the self-attention computation, H denotes the number of multi-head self-attention heads, and $\boldsymbol{Q}_i$, $\boldsymbol{K}_i$, and $\boldsymbol{V}_i$ of the i-th head are obtained by the following linear projection:

$$\boldsymbol{Q}_i = \boldsymbol{X}_i \boldsymbol{W}_i^Q \tag{11}$$

$$\boldsymbol{K}_i = \mathrm{AvgPool}(\boldsymbol{X}_i)\boldsymbol{W}_i^K \tag{12}$$

$$\boldsymbol{V}_i = \mathrm{AvgPool}(\boldsymbol{X}_i)\boldsymbol{W}_i^V \tag{13}$$

Equations (11)–(13) describe that $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ are obtained by multiplying the input feature map with three different weight matrices. AvgPool denotes the application of pooling operations on the input feature map to reduce spatial dimensions, thereby decreasing the computational load and complexity while preserving important feature information.

The feed forward network (FFN) is crucial in each attention block of the Transformer and is responsible for the nonlinear mapping and transformation of the features of the image patch, thereby improving the representation ability and performance of the model. In the original vision Transformer [16], the FFN consists of two fully connected layers, and

a GELU activation function is included between the two layers, which can be expressed as follows:

$$\mathrm{FFN}(\boldsymbol{X}) = \mathrm{GELU}(\boldsymbol{X}W_1 + b_1)W_2 + b_2 \tag{14}$$

where W_1 and W_2 denote the weights of the two linear layers, and b_1 and b_2 indicate the bias terms, respectively.

In this work, we proposed an improved feed forward network (IFFN), as shown in Figure 7, which can further improve the representation capability of the network. The IFFN mainly consists of a point-wise convolution, a depth-wise convolution, channel attention [27], a projection layer, and the necessary residual connections. The IFFN incorporates depth-wise convolution and channel attention, where depth-wise convolution is used to extract local information at a negligible computational cost, while channel attention adaptively learns the weights of each channel of the feature map, thus enhancing important feature channels and suppressing unimportant ones. Compared with a typical FFN, the IFFN can better capture the local and long-range dependencies of the feature map, thereby improving the performance of the model. And the IFFN can be formulated as follows:

$$\mathrm{IFFN}(\boldsymbol{X}) = f_{1\times1}(\boldsymbol{\mathcal{F}}(f_{1\times1}(\boldsymbol{X}))) + \boldsymbol{X} \tag{15}$$

$$\boldsymbol{\mathcal{F}}(\boldsymbol{X}) = \mathrm{CA}(\mathrm{ReLU}(\mathrm{DWConv}(\boldsymbol{X}))) + \boldsymbol{X} \tag{16}$$

where CA denotes channel attention [27], DWConv denotes 3×3 depth-wise convolution, $f_{1\times1}$ denotes pointwise convolution, and GELU is a nonlinear activation function.

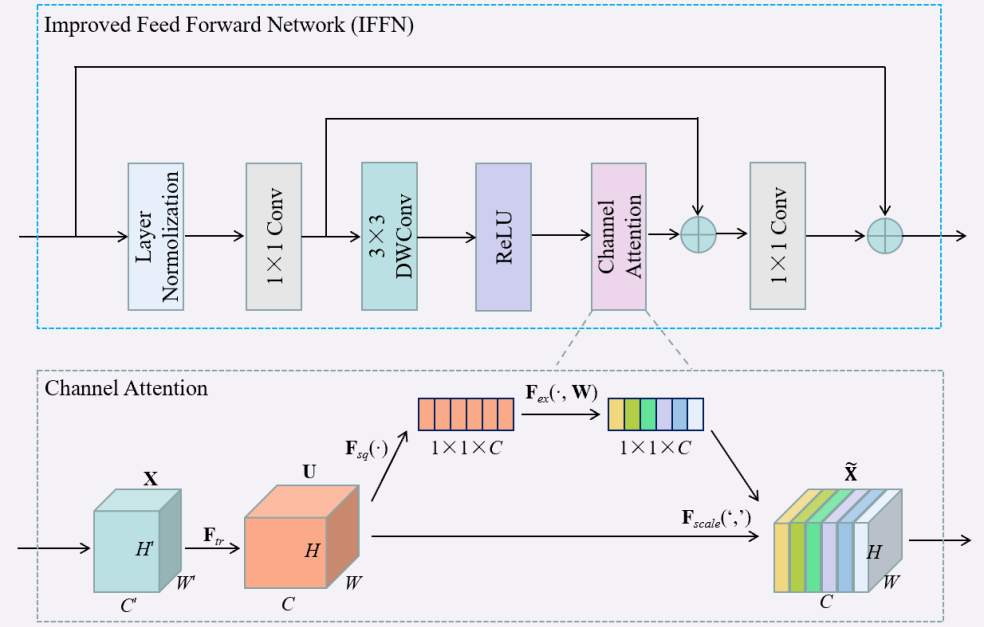

Figure 7. A schematic diagram of the IFFN.

The GFEB utilizes the multi-head self-attention mechanism to capture global semantic information and uses the constructed IFFN to better capture the local and remote dependencies of the feature map, providing a richer and more accurate feature representation for the model, which greatly improves the model's performance.

With the aforementioned parts, we propose an innovative hybrid network, HFE-Net, based on a CNN and vision Transformer, which utilizes Transformer to capture long-range dependencies, while a CNN is used to extract local information, thus enhancing the feature representation capability of the model. And the detailed structures of each module of the HFE-Net are shown in Table 1.

Table 1. Detailed structures of each component of the proposed HFE-Net.

Operator	Input Size	Number	Stride
Conv2D	224 × 224 × 3	1	2
Conv2D	112 × 112 × 36	1	1
Conv2D	112 × 112 × 24	2	2, 1
PatchEmbed	56 × 56 × 36	1	-
LFEB	56 × 56 × 48	3	1
GFEB	56 × 56 × 48	1	1
PatchEmbed	56 × 56 × 48	1	-
LFEB	28 × 28 × 96	4	1
GFEB	28 × 28 × 96	1	1
PatchEmbed	28 × 28 × 96	1	-
LFEB	14 × 14 × 192	8	1
GFEB	14 × 14 × 240	1	1
PatchEmbed	14 × 14 × 240	1	-
LFEB	7 × 7 × 384	3	1
GFEB	7 × 7 × 384	1	1
Avg Pool	7 × 7 × 384	1	-
FC	1 × 1 × 384	1	-

2.4. Efficient Multi-Scale Feature Fusion Module (EMFF)

For object detection tasks, multi-scale feature fusion is crucial for locating and identifying objects at different scales. Although a typical FPN [28] and PANet [29] have achieved remarkable results in processing multi-scale features, they still face some challenges when facing some complex scenes and small-scale objects. For example, for the fine-grained cocoon recognition task of this work, due to the similar appearance between cocoon categories and the unapparent defects in the inferior cocoons, which results in insufficient feature information to differentiate the cocoon categories, it is difficult to accurately and efficiently accomplish the fine-grained cocoon detection task.

To address the above problems, we propose a novel and efficient multi-scale feature fusion module (EMFF), which adopts the same design paradigm as the PANet [29], except we construct a novel downsampling (by halving the feature map spatial resolution and doubling the channels) module when dealing with downsampling, as shown in Figure 8. In contrast to the conventional 3 × 3 convolution with a stride of 2, which is commonly employed for downsampling, the optimized downsampling module we propose incorporates a combination of diverse downsampling techniques. This approach is designed to mitigate the loss of fine-grained information, ensuring that the model retains the necessary details for accurate feature representation. Specifically, for the feature map input to the downsampling module, we initially apply a 1 × 1 convolution to generate three separate copies. Subsequently, we employ a slicing-based downsampling technique on the first copy, which is designed to preserve the original spatial details of the feature map. The second one uses the max-pooling downsampling method to retain the key information of the feature map and avoid the loss of fine-grained features. The third one uses a 3 × 3 convolution with a stride of 2 to extract local features while downsampling. Finally, the downsampling results obtained by these three different downsampling methods are concatenated, and a point-wise convolution is used to fuse and reduce the dimension.

The optimized downsampling module constructed in this work obtains complementary feature maps by using three different downsampling methods: max-pooling, convolution with a stride of 2, and cut-slice. The fusion of these complementary downsampled feature maps enables the network to retain more fine-grained information during downsampling and avoid the loss of key features, thus improving the accuracy of fine-grained cocoon detection.

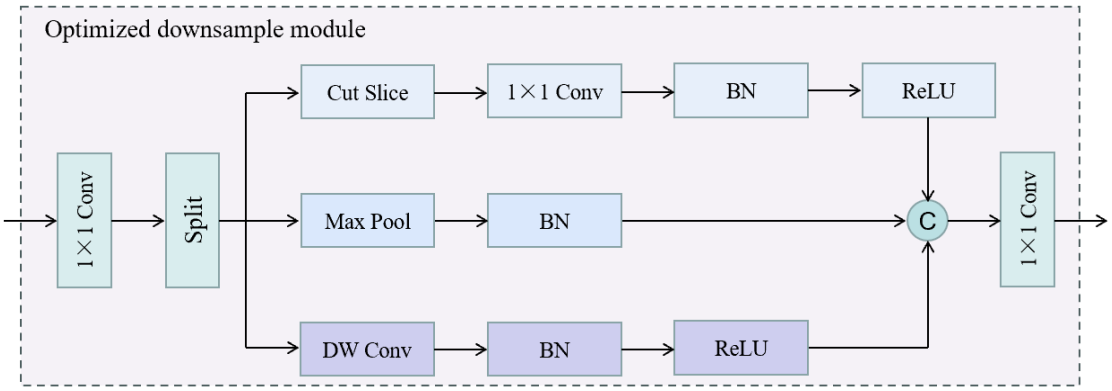

Figure 8. A schematic diagram of the optimized downsampling module.

2.5. Overall Framework of the Model

An efficient and accurate model design is imperative for real-time object detection. In this work, we propose AMMF-Net, an object detection model based on attention mechanism and multi-scale feature fusion, which aims to address the limitations of typical object detectors in multi-scale fine-grained feature extraction and fusion to improve the recognition accuracy of cocoon detection. Building on the classical single-stage object detector, RetinaNet [30], we modify it in two aspects: firstly, we propose a more efficient feature extraction network, which combines the advantages of both CNN and Transformer paradigms of feature extraction networks; secondly, we design an efficient multi-scale feature fusion module to replace the original neck part of RetinaNet so as to better perform multi-scale feature fusion and avoid the loss of fine-grained information in the fusion process. These two parts correspond to HFE-Net, introduced in Section 2.3, and EMFF, introduced in Section 2.4, respectively.

Based on the constructed HFE-Net and EMFF, combined with the detection head of RetinaNet, we propose a novel object detection model, AMMF-Net, which is used for fine-grained cocoon recognition, and its network structure is illustrated in Figure 9.

2.6. Implementation Details

The model is developed based on the Python and Pytorch frameworks. The models were trained on a NVIDIA (2788 San Tomas Expressway, Santa Clara, CA, USA) RTX 3090 GPU with 24G of memory. For model optimization, we used the Adam [31] optimizer with a weight decay of 0.0001 and the number of iterations of the model was 100 epochs. The batch sizes for the feature extraction network HFE-Net and the object detection network AMMF-Net are 64 and 12, respectively. The feature extraction network HFE-Net selected the cross-entropy loss as the loss function, which is expressed as follows:

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{M} l_{i,c}\log(p_{i,c}) \tag{17}$$

where N denotes the amount of data in the dataset and M denotes the number of categories. $l_{i,c}$ represents the true label for the i-th sample regarding the c-th class, and $p_{i,c}$ signifies the probability assigned by the model to the i-th sample belonging to the c-th class.

The experiment takes the prediction accuracy (acc) as the evaluation metric of the feature extraction network HFE-Net, which can be calculated using the following formula:

$$acc = \frac{TP + TN}{TP + FP + TN + FN} \tag{18}$$

where TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively, i.e., "predicted as a positive sample and correctly predicted", "predicted as a negative sample and correctly predicted", "predicted as a positive sample but

incorrectly predicted", and "predicted as a negative sample but incorrectly predicted", respectively.

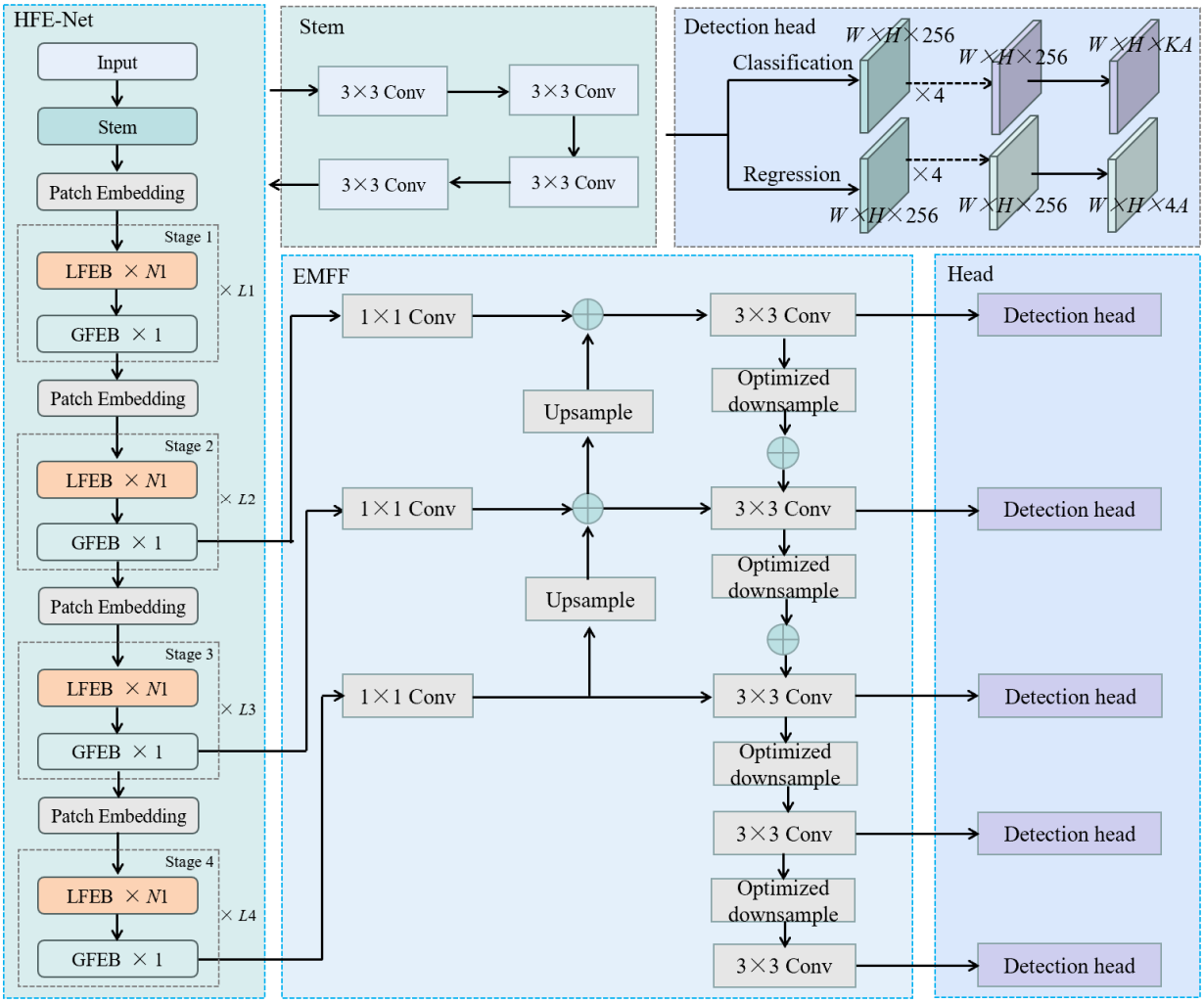

Figure 9. A schematic diagram of the proposed AMMF-Net.

For the object detection network AMMF-Net, we use mAP as the evaluation metric, which can be calculated using the following formula:

$$\text{mAP} = \frac{1}{k}\sum_{i=1}^{k} AP_i \tag{19}$$

$$\text{AP} = \int_0^1 P(R)\, dR \tag{20}$$

$$\text{P} = \frac{TP}{TP + FP} \tag{21}$$

$$\text{R} = \frac{TP}{TP + FN} \tag{22}$$

where P denotes precision, which measures the proportion of samples predicted by the model to be positive instances that are actually positive instances, and R denotes recall, which measures the model's ability to successfully predict all positive instances. Here, we use the IoU threshold of 0.5 to divide TP and FP. AP denotes average precision, which measures the object detection accuracy of the model on each category, and it can be derived by calculating the area under the precision–recall curve. k denotes the number of

predicted object categories, and mAP synthesizes the object detection accuracy for these k different categories.

3. Results and Discussion

3.1. Performance of Designed AMMF-Net

Considering that the latest object detectors have extensively applied complex optimization techniques, we chose the classical one-stage object detector RetinaNet, whose backbone is ResNet-50, as the benchmark model. To improve the performance of the model, we refined RetinaNet by replacing its backbone with our own proposed novel and efficient feature extraction network, HFE-Net, and meanwhile, we replaced the FPN of RetinaNet with our optimized efficient multi-scale feature fusion module, EMFF, to form an object detection model suitable for the fine-grained identification of silkworm cocoons, i.e., AMMF-Net. Additionally, to comprehensively evaluate the performance of AMMF-Net, several other advanced object detectors were used for comparison experiments to clearly demonstrate the effectiveness of the proposed model.

Table 2 shows the performance of the proposed AMMF-Net on the constructed cocoon dataset. Compared to other advanced object detectors, the proposed model achieves superior prediction accuracies for the identification of both normal cocoons (normal) and defective cocoons (inferior), which are 69.75% and 62.48%, respectively. This indicates that AMMF-Net is able to effectively distinguish different types of cocoons. In particular, the mAP value of 66.12% achieved by AMMF-Net is not only the highest among all models participating in the comparison, but also 2.13% higher than the baseline RetinaNet. This improvement highlights the effectiveness of the efficient feature extraction network and multi-scale fusion module constructed in this article, allowing the model to more accurately locate and identify objects.

Table 2. Comparison of cocoon detection results between AMMF-Net constructed in this work and other models.

Method	Normal			Inferior			mAP
	P	R	AP	P	R	AP	
YOLOv3 [32]	83.51	66.20	68.13	90.92	52.86	59.62	63.87
YOLOv5-l [33]	82.37	66.17	68.13	91.57	50.50	58.48	63.30
YOLOX-l [34]	84.44	66.24	68.51	89.59	54.05	60.38	64.44
YOLOv7-l [35]	85.68	66.08	68.65	84.92	55.98	60.70	64.68
Faster RCNN [36]	87.56	65.73	68.70	91.13	51.74	59.54	64.12
Cascade RCNN [37]	84.74	65.30	68.20	85.77	54.70	60.18	64.19
RetinaNet (baseline) [30]	81.95	66.26	68.13	90.90	52.91	59.85	63.99
AMMF-Net (ours)	89.86	66.67	69.75	90.44	57.39	62.48	66.12

To more comprehensively evaluate the performance of the models, this paper also provides an experimental comparison of the computational complexity of these methods. In the field of object detection, the computational complexity is usually measured by the number of parameters (Params) and the number of floating-point operations per second (GFLOPs) of the model, which directly affect the inference speed and hardware requirements of the model. In Table 3, we compare the mAP, number of parameters, and GFLOPs of AMMF-Net with several other advanced object detection models. It can be seen that the proposed AMMF-Net not only achieves the best detection of mulberry cocoons, but also realizes the least number of parameters. Regarding the computational complexity of the model, although AMMF-Net is more computationally intensive than YOLOv7, its detection accuracy is 1.44% higher than that of YOLOv7. This indicates that AMMF-Net has achieved the optimal speed–accuracy tradeoff by taking into account the computational efficiency of the model while pursuing high accuracy.

Table 3. Comparison of computational complexity between AMMF-Net and other object detectors.

Model	mAP	Params	GFLOPs
YOLOv3	63.87	61.95	156.62
YOLOv5-l	63.30	46.73	144.89
YOLOX-l	64.44	54.21	156.01
YOLOv7-l	64.68	37.62	106.47
Faster RCNN	64.12	41.22	182.23
Cascade RCNN	64.19	69.17	238.10
RetinaNet (baseline)	63.99	37.74	170.21
AMMF-Net (ours)	66.12	21.33	135.40

To more intuitively demonstrate the effectiveness of the proposed method, we visualize the detection results of the baseline (RetinaNet) and AMMF-Net on the constructed mulberry cocoon dataset, as shown in Figure 10. Specifically, Figure 10a–d show the detection results of the baseline, and the corresponding (a'), (b'), (c'), and (d') are the detection results of the proposed AMMF-Net. As shown in Figure 10a–c, the baseline method suffers from an incorrect prediction of the categories of some cocoons (marked by yellow arrows). The main reasons for this are that cocoon detection is a fine-grained recognition issue, the difference between cocoon categories is small, and there are some inferior cocoons in which the defects occupy little pixels, resulting in it being difficult for the general object detection algorithms to extract adequate fine-grained feature information. In addition, as shown in Figure 10d, for some dense and occluded scenes, the baseline cannot accurately detect all objects. Comparatively, the proposed AMMF-Net model can accurately detect all objects, as shown in Figure 10a'–d'. The comparative analysis of the visual results underscores the strengths of AMMF-Net in handling the unique challenges posed by cocoon detection. The model's performance is particularly noteworthy given the small differences between cocoon categories and the presence of occlusions.

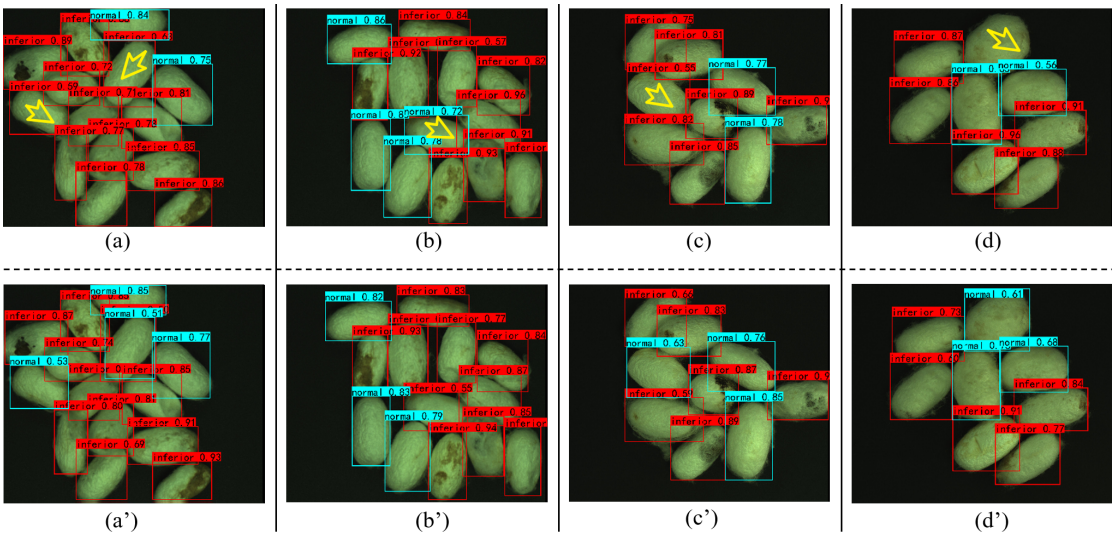

Figure 10. The visual results of the proposed AMMF-Net for cocoon detection. (**a**–**d**) are the detection results of the baseline, while the corresponding (**a'**–**d'**) are the detection results of the proposed AMMF-Net. The arrows indicate cocoons that were incorrectly predicted or not detected by the RetinaNet model.

Furthermore, the variation in loss values during the training of AMMF-Net is shown in Figure 11. It can be observed that as the network is trained, both the training loss and validation loss gradually decrease and then converge to a small range of values. This phenomenon indicates that the model has not only learned effective feature representations on the training set but also possesses good generalization capabilities, achieving low error

rates on the unseen validation set. The convergence characteristic of the model suggests that it has avoided overfitting, meaning it has not merely memorized specific samples from the training set but has learned general patterns that can be applied to new data.

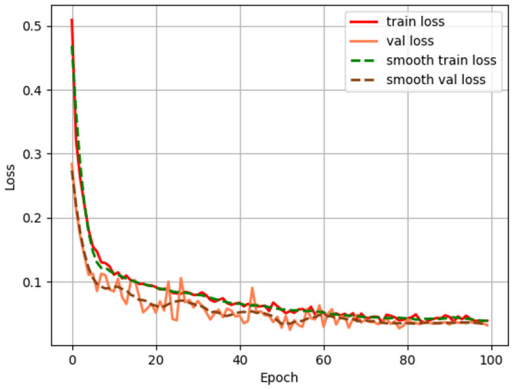

Figure 11. Loss curves during training.

To more comprehensively evaluate the performance of the model, apart from using evaluation metrics such as mAP, GFLOPs, and Params, we also depicted the precision–recall curves for AMMF-Net and the benchmark model on the testing set, as shown in Figure 12. Figure 12a,b depict the detection performance of the proposed method, AMMF-Net, for inferior and normal silkworm cocoons, respectively. In contrast, Figure 12a′,b′ illustrate the detection performance of the baseline method, RetinaNet, on the same categories of cocoons.

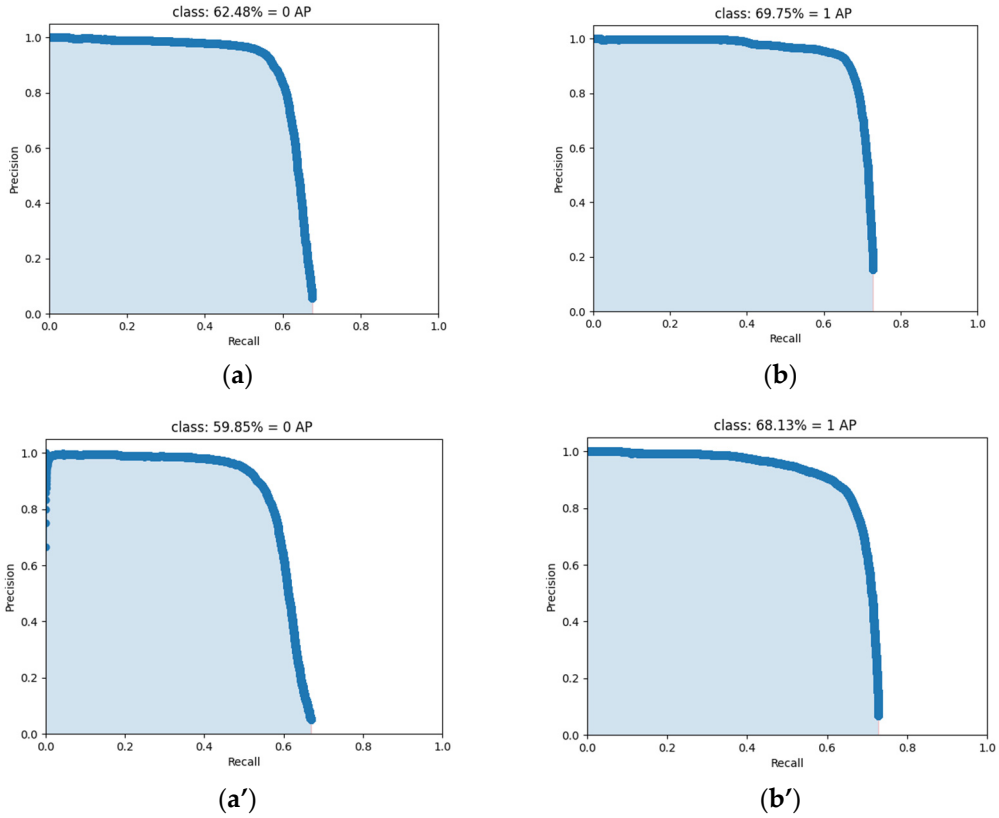

Figure 12. (**a**) PR curve of AMMF-Net for inferior cocoons, (**b**) PR curve of AMMF-Net for normal cocoons, (**a′**) PR curve RetinaNet for inferior cocoons, and (**b′**) PR curve of RetinaNet for normal cocoons.

The precision–recall (PR) curves within these figures reflect the relationship between precision and recall at various confidence threshold levels, providing a comprehensive evaluation of the models' performance across the spectrum of classification certainty. The area under the PR curve (AP) serves as a measure of a model's ability to balance precision and recall, and it is a critical indicator of performance across different classes of data. A higher AP value signifies superior model performance on a particular category. It is noteworthy that AMMF-Net achieves an improvement of 1.62% in AP for normal cocoons and a more substantial enhancement of 2.63% in AP for defective cocoons compared to the baseline RetinaNet method. This increase in the AP value demonstrates the superior performance of AMMF-Net in terms of its ability to accurately detect both normal and defective cocoons, thereby offering a more effective solution for fine-grained cocoon recognition tasks.

To more comprehensively evaluate the performance of AMMF-Net and demonstrate its generalization performance, we conducted further experiments on the publicly available dataset VOC07 + 12. Table 4 summarizes the detection results of AMMF-Net with other advanced object detectors on this dataset. From Table 4, it is evident that among all the methods compared, AMMF-Net exhibits the most compact model size in terms of parameter count. Moreover, despite having a slightly higher computational load compared to YOLOv7, AMMF-Net achieves a 1.6% higher mAP score. Additionally, when evaluating detection performance on this dataset, although AMMF-Net is marginally lower by 0.6% compared to Faster R-CNN, it is important to note that Faster R-CNN is a two-stage object detection algorithm with significantly higher computational and parameter demands than the method presented in this paper. Therefore, in a comprehensive assessment that takes into account both accuracy and efficiency, AMMF-Net delivers superior performance. Thus, in a comprehensive view, the experimental results strongly favor the superiority of the proposed model. The proposed model not only has excellent performance in the fine-grained cocoon detection task, but also shows significant competitive advantages in dealing with the generalized object detection issue. This provides a solid foundation for the wide applicability of AMMF-Net in practical applications, which also has positive significance in advancing the research and applications in the field of general object detection.

Table 4. Comparison of detection results of proposed AMMF-Net and other advanced object detectors on VOC07 + 12.

Model	Params (MB)	GFLOPs	mAP (%)
Faster RCNN	41.22	182.23	78.2
Cascade RCNN	69.17	238.10	77.2
RetinaNet	37.74	170.21	75.8
YOLOv3	61.95	156.62	73.3
YOLOv5-l	46.73	144.89	76.5
YOLOX-l	54.21	156.01	75.1
YOLOv7-l	37.62	106.47	76.0
AMMF-Net (ours)	21.33	135.40	77.6

3.2. Ablation Study

For a deeper understanding of the proposed AMMF-Net and to validate the effectiveness of its key designs, we conducted a series of ablation experiments. Specifically, we designed two ablation experiments, including one concerned with the proposed efficient feature extraction network, HFE-Net, which is performed on the ImageNet-100 dataset. The other ablation experiment focuses on the designed efficient multi-scale feature fusion module, EMFF, which is performed on the constructed mulberry cocoon dataset as well as the publicly available dataset VOC07 + 12. These experiments aim to provide insights into the independent contributions of the components of our proposed approach and their generalization performance on different tasks and datasets.

Ablation study of proposed HFE-Net. In order to verify the superiority of the proposed HFE-Net, a feature extraction network that combines the advantages of a CNN and

Transformer, we compare the experimental results with those of other advanced feature extraction networks [12,16,18,25,38–41] such as ResNet, Swin Tranformer, and PVT on the publicly available dataset ImageNet-100. The experimental results are shown in Figure 13, where HFE-Net achieves comparable prediction results with other state-of-the-art feature extraction networks, with a prediction accuracy of 77.3%. However, its parameters and computational cost are only 11.79 MB and 2.18 GMACs, respectively, which shows that it significantly outperforms other networks and achieves the best accuracy–speed trade-off.

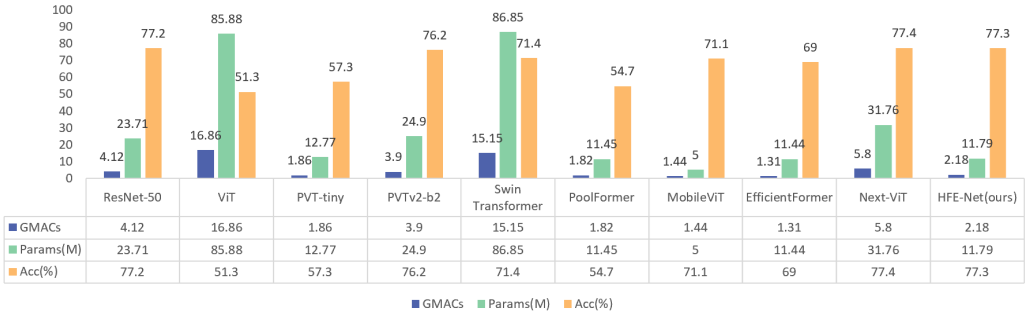

Figure 13. Performance comparison of proposed HFE-Net with other models on ImageNet-100.

Ablation study of the proposed EMFF. Furthermore, we provide an insight into the impact of the proposed multi-scale feature fusion module (EMFF) on the object detection results. The experiments are conducted on our optimized RetinaNet architecture in which the backbone adopts our proposed highly efficient feature extraction network, HFE-Net. The experimental results are shown in Table 5, which demonstrates that EMFF achieves 1.93% and 0.8% improvements in detection performance on the silkworm cocoon dataset and the VOC07 + 12 public dataset, respectively, without significantly increasing the computational cost and parameters, compared to the classical multi-scale feature fusion module, FPN. Therefore, it not only verifies the significant performance improvement of the proposed EMFF for multi-scale feature fusion for fine-grained cocoon recognition, but also shows its wide applicability in different scenarios.

Table 5. Ablation experimental results of proposed efficient multi-scale feature fusion module (EMFF).

Dataset	Method	Params (MB)	GFLOPs	mAP (%)
Cocoon	RetinaNet	37.74	170.21	63.99
	RetinaNet + HFE-Net	20.03	129.12	64.19
	RetinaNet + HFE-Net + EMFF(AMMF-Net)	21.33	135.40	66.12
VOC07 + 12	RetinaNet	37.74	170.21	75.8
	RetinaNet + HFE-Net	20.03	129.12	76.8
	RetinaNet + HFE-Net + EMFF	21.33	135.40	77.6

4. Conclusions

To address the problems of small differences between cocoon categories, the small number of pixels occupied by defects in inferior cocoons, and an insufficient amount of feature information that can distinguish cocoon categories, we designed AMMF-Net, a model for fine-grained cocoon detection based on the attention mechanism and multi-scale feature fusion. Firstly, we proposed a novel hybrid model HFE-Net based on vision Transformer and CNN networks for the efficient extraction of fine-grained cocoon image features. Secondly, we designed a novel multi-scale feature fusion module, EMFF, for improving the information loss problem of existing multi-scale feature fusion methods, which effectively improved the detection accuracy of fine-grained silkworm cocoon recognition. Finally, based on the constructed efficient fine-grained feature extraction network, HFE-Net, and the optimized multi-scale feature fusion module, EMFF, the object detection algorithm,

AMMF-Net, for cocoon sorting was constructed. Extensive experiments demonstrated that the constructed object detection algorithm achieves the best experimental results.

Although the detection results of the proposed AMMF-Net are better than other existing object detection networks, and some progress has been made in the fine-grained cocoon recognition method, the proposed method still suffers from the problem of leakage detection. Therefore, we will further explore the following aspects in our future work: (1) We will expand the dataset to make it more comprehensive, covering as many real-world sorting scenarios as possible to enhance the model's generalization capability. (2) We will further optimize and improve the rotation object detection algorithm to reduce false negatives and false positives, thereby enhancing the performance and efficiency of the algorithm. Additionally, we plan to integrate more modalities of data, such as sensor data, to build a multimodal data fusion model, aiming to improve the detection and classification accuracy of silk cocoon sorting, refine cocoon categories, and provide greater value to the silk industry.

Author Contributions: All authors contributed to this study's conception and design. Material preparation, data collection, and analysis were performed by H.Z., Y.M. and X.Z. The algorithm design and implementation of the model was mainly carried out by X.G. and J.C. The first draft of the manuscript was written by T.Z. and all authors commented on previous versions of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This paper is sponsored by the National Study Abroad Fund of China and Key Laboratory of AI and Information Processing (Hechi University), the Education Department of Guangxi Zhuang Autonomous Region (2022GXZDSY001), and the 2023 Basic Research Ability Enhancement Project for Young and Middle age Teachers in Universities of Guangxi (2023KY0632).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The mulberry cocoon dataset generated and analyzed during the current study are available from the corresponding author upon reasonable request. And the other data relevant to this research are available in [22,23].

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Wen, C.; Wen, J.; Li, J.; Luo, Y.; Chen, M.; Xiao, Z.; Xu, Q.; Liang, X.; An, H. Lightweight Silkworm Recognition Based on Multi-Scale Feature Fusion. *Comput. Electron. Agric.* **2022**, *200*, 107234. [CrossRef]
2. Nahiduzzaman, M.; Chowdhury, M.E.H.; Salam, A.; Nahid, E.; Ahmed, F.; Al-Emadi, N.; Ayari, M.A.; Khandakar, A.; Haider, J. Explainable Deep Learning Model for Automatic Mulberry Leaf Disease Classification. *Front. Plant. Sci.* **2023**, *14*, 1175515. [CrossRef]
3. Xiong, H.; Cai, J.; Zhang, W.; Hu, J.; Deng, Y.; Miao, J.; Tan, Z.; Li, H.; Cao, J.; Wu, X. Deep Learning Enhanced Terahertz Imaging of Silkworm Eggs Development. *iScience* **2021**, *24*, 103316. [CrossRef] [PubMed]
4. Wang, Q.; Li, Z.; Gu, T.; Ye, F.; Wang, X. Cocoons Counting and Classification Based on Image Processing. In Proceedings of the 2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Chengdu, China, 17–19 October 2020; pp. 148–152.
5. Guo, F.; He, F.; Tao, D.; Li, G. Automatic Exposure Correction Algorithm for Online Silkworm Pupae (Bombyx Mori) Sex Classification. *Comput. Electron. Agric.* **2022**, *198*, 107108. [CrossRef]
6. Sumriddetchkajorn, S.; Kamtongdee, C.; Chanhorm, S. Fault-Tolerant Optical-Penetration-Based Silkworm Gender Identification. *Comput. Electron. Agric.* **2015**, *119*, 201–208. [CrossRef]
7. Tao, D.; Wang, Z.; Li, G.; Qiu, G. Radon Transform-Based Motion Blurred Silkworm Pupa Image Restoration. *Int. J. Agric. Biol. Eng.* **2019**, *12*, 152–159. [CrossRef]
8. Cai, J.; Yuan, L.; Liu, B.; Sun, L. Nondestructive Gender Identification of Silkworm Cocoons Using X-Ray Imaging with Multivariate Data Analysis. *Anal. Methods* **2014**, *6*, 7224–7233. [CrossRef]
9. Vasta, S.; Figorilli, S.; Ortenzi, L.; Violino, S.; Costa, C.; Moscovini, L.; Tocci, F.; Pallottino, F.; Assirelli, A.; Saviane, A.; et al. Automated Prototype for Bombyx Mori Cocoon Sorting Attempts to Improve Silk Quality and Production Efficiency through Multi-Step Approach and Machine Learning Algorithms. *Sensors* **2023**, *23*, 868. [CrossRef] [PubMed]
10. Yang, C.; Peng, J.; Cai, J.; Tang, Y.; Zhou, L.; Yan, Y. Research and Design of a Machine Vision-Based Silk Cocoon Quality Inspection System. In Proceedings of the 2023 IEEE 10th International Conference on Cyber Security and Cloud Computing

(CSCloud)/2023 IEEE 9th International Conference on Edge Computing and Scalable Cloud (EdgeCom), Xiangtan, China, 1–3 July 2023; pp. 369–374.
11. Li, S.; Sun, W.; Liang, M.; Shao, T. Research on the Identification Method of Silkworm Cocoon Species Based on Improved YOLOv3. In Proceedings of the 2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Harbin, China, 25–27 December 2020; pp. 1119–1123.
12. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
13. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
14. Tan, M.; Le, Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 24 May 2019; pp. 6105–6114.
15. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2017; Volume 30.
16. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv:2010.11929.
17. Jiang, Z.-H.; Hou, Q.; Yuan, L.; Zhou, D.; Shi, Y.; Jin, X.; Wang, A.; Feng, J. All Tokens Matter: Token Labeling for Training Better Vision Transformers. In Proceedings of the Advances in Neural Information Processing Systems; Curran Associates, Inc.: New York, NY, USA, 2021; Volume 34, pp. 18590–18602.
18. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.
19. Dong, X.; Bao, J.; Chen, D.; Zhang, W.; Yu, N.; Yuan, L.; Chen, D.; Guo, B. CSWin Transformer: A General Vision Transformer Backbone with Cross-Shaped Windows. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 12124–12134.
20. Li, G.; Xu, D.; Cheng, X.; Si, L.; Zheng, C. SimViT: Exploring a Simple Vision Transformer with Sliding Windows. In Proceedings of the 2022 IEEE International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, 18–22 July 2022; pp. 1–6.
21. Maaz, M.; Shaker, A.; Cholakkal, H.; Khan, S.; Zamir, S.W.; Anwer, R.M.; Shahbaz Khan, F. EdgeNeXt: Efficiently Amalgamated CNN-Transformer Architecture for Mobile Vision Applications. In Proceedings of the Computer Vision—ECCV 2022 Workshops; Karlinsky, L., Michaeli, T., Nishino, K., Eds.; Springer: Cham, Switzerland, 2023; pp. 3–20.
22. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Kai, L.; Li, F.-F. ImageNet: A Large-Scale Hierarchical Image Database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 22–24 June 2009; pp. 248–255.
23. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
24. Xiao, T.; Singh, M.; Mintun, E.; Darrell, T.; Dollar, P.; Girshick, R. Early Convolutions Help Transformers See Better. In Proceedings of the Advances in Neural Information Processing Systems; Curran Associates, Inc.: New York, NY, USA, 2021; Volume 34, pp. 30392–30400.
25. Li, J.; Xia, X.; Li, W.; Li, H.; Wang, X.; Xiao, X.; Wang, R.; Zheng, M.; Pan, X. Next-ViT: Next Generation Vision Transformer for Efficient Deployment in Realistic Industrial Scenarios. *arXiv* **2022**, arXiv:2207.05501.
26. Xu, B.; Wang, N.; Chen, T.; Li, M. Empirical Evaluation of Rectified Activations in Convolutional Network. *arXiv* **2015**, arXiv:1505.00853.
27. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
28. Lin, T.-Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
29. Wang, K.; Liew, J.H.; Zou, Y.; Zhou, D.; Feng, J. PANet: Few-Shot Image Semantic Segmentation with Prototype Alignment. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October 2019–2 November 2019; pp. 9197–9206.
30. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
31. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2017**, arXiv:1412.6980.
32. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
33. Jocher, G.; Stoken, A.; Borovec, J.; NanoCode012; ChristopherSTAN; Changyu, L.; Laughing; Hogan, A.; Lorenzomammana; Tkianai; et al. *Zenodo, Ultralytics/Yolov5: V3.0*; Zenodo: Geneva, Switzerland, 2020. [CrossRef]
34. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
35. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.

36. 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]
37. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
38. Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. PVT v2: Improved Baselines with Pyramid Vision Transformer. *Comp. Visual Media* **2022**, *8*, 415–424. [CrossRef]
39. Li, Y.; Yuan, G.; Wen, Y.; Hu, J.; Evangelidis, G.; Tulyakov, S.; Wang, Y.; Ren, J. EfficientFormer: Vision Transformers at MobileNet Speed. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 12934–12949.
40. Mehta, S.; Rastegari, M. Mobilevit: Light-Weight, General-Purpose, And Mobile-Friendly Vision Transformer. *arXiv* **2022**, arXiv:2110.02178.
41. Yu, W.; Luo, M.; Zhou, P.; Si, C.; Zhou, Y.; Wang, X.; Feng, J.; Yan, S. MetaFormer Is Actually What You Need for Vision. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 10809–10819.