

## Article

# Research on Defect Detection in Automated Fiber Placement Processes Based on a Multi-Scale Detector

Yongde Zhang <sup>1,2</sup>, Wei Wang <sup>2,\*</sup> , Qi Liu <sup>2</sup>, Zhonghua Guo <sup>2</sup> and Yangchun Ji <sup>2</sup>

<sup>1</sup> Key Laboratory of Advanced Manufacturing Technology of Zhejiang Province, College of Mechanical Engineering, Zhejiang University, Hangzhou 310027, China

<sup>2</sup> Avic Xi'an Aircraft Industry Group Company Ltd., Xi'an 710089, China

\* Correspondence: wangwei\_4524@163.com; Tel.: +86-178-0115-2471

**Abstract:** Various surface defects in automated fiber placement (AFP) processes affect the forming quality of the components. In addition, defect detection usually requires manual observation with the naked eye, which leads to low production efficiency. Therefore, automatic solutions for defect recognition have high economic potential. In this paper, we propose a multi-scale AFP defect detection algorithm, named the spatial pyramid feature fusion YOLOv5 with channel attention (SPFFY-CA). The spatial pyramid feature fusion YOLOv5 (SPFFY) adopts spatial pyramid dilated convolutions (SPDCs) to fuse the feature maps extracted in different receptive fields, thus integrating multi-scale defect information. For the feature maps obtained from a concatenate function, channel attention (CA) can improve the representation ability of the network and generate more effective features. In addition, the sparsity training and pruning (STP) method is utilized to achieve network slimming, thus ensuring the efficiency and accuracy of defect detection. The experimental results of the PASCAL VOC and our AFP defect datasets demonstrate the effectiveness of our scheme, which achieves superior performance.

**Keywords:** AFP defect detection; multi-scale object detection; model compression; convolution neural network



**Citation:** Zhang, Y.; Wang, W.; Liu, Q.; Guo, Z.; Ji, Y. Research on Defect Detection in Automated Fiber Placement Processes Based on a Multi-Scale Detector. *Electronics* **2022**, *11*, 3757. <https://doi.org/10.3390/electronics11223757>

Academic Editors: D.J. Lee and Dong Zhang

Received: 22 October 2022

Accepted: 11 November 2022

Published: 16 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



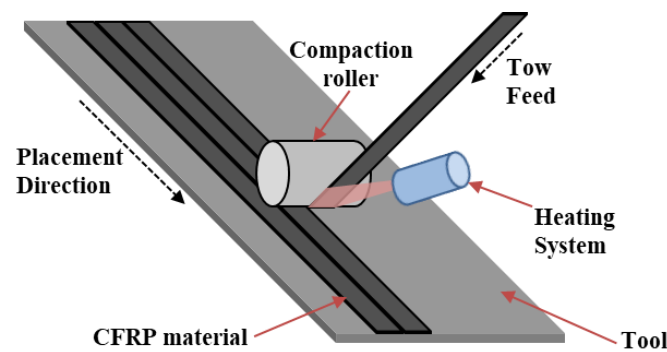
**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

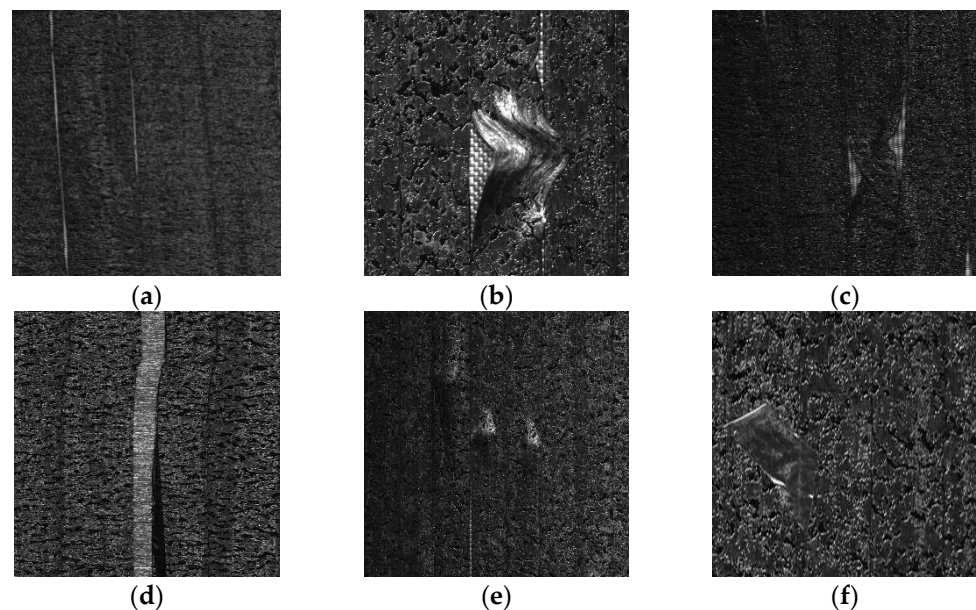
Carbon fiber-reinforced plastic (CFRP) has remarkable advantages such as light weight, high strength, fatigue resistance, and corrosion resistance, and it is often used in large, single-piece aircraft structures [1,2]. The manufacturing methods of CFRP include hand layup, automated tape laying, and automated fiber placement [3]. Given the problems associated with the hand layup process, which include difficulty in achieving complex shapes, the need for the manufacturing of large-sized parts, low efficiency, and difficulty in achieving quality consistency, the relatively novel technique of automated fiber placement (AFP) is increasingly used in industry to make manufacturing economical, fast, and efficient [4–6]. Automated fiber placement (AFP) contains a placement head and a robotic arm. The placement head lays the CFRP material layer by layer onto a mold. The procedure of automatic fiber laying (AFP) is schematically shown in Figure 1.

In the actual production environment, various defects may occur during fiber layup, which will affect the quality [7–10]. These defects are often directly related to the layup process itself. Harik et al. [7] investigated the link between AFP defects and process planning, layup strategies, and machining.

The common types of AFP defects include wrinkles, twists, gaps, bubbles, and the presence of foreign material. A series of scanned images of actual defects and a reference sample without any defect are illustrated in Figure 2.



**Figure 1.** The automated fiber placement (AFP) working principle.



**Figure 2.** Common manufacturing defects in the AFP process: (a) no defects, (b) a wrinkle, (c) a twist, (d) a gap, (e) bubbles, and (f) foreign material.

Defect detection typically requires manual observation by the naked eye, which leads to low production efficiency. Manual online detection is easily affected by subjective experience, and it may cause problems such as missed detection when the manufacturing task is heavy. With the rapid development of computer vision, deep learning, and other technologies, the defect visual inspection technology [11–15] based on deep learning can be effectively used in the quality control and monitoring of the CFRP manufacturing process. Sebastian Zambal et al. [8] considered defect detection in AFP as an image segmentation problem that can be trained by manually generated training sets. In their study, a laser triangulation sensor was used to obtain the data of the layup machinery, and a dataset with 5000 samples was established. The trained neural network could recognize the gaps, overlaps, and foreign objects on the product's surface.

In this paper, we propose the spatial pyramid feature fusion YOLOv5 with channel attention (SPFFY-CA) to achieve defect detection in AFP. The SPFFY-CA includes spatial pyramid dilated convolutions (SPDCs) and channel attention (CA) modules. In addition, we used the sparsity training and pruning (STP) method to achieve network slimming and ensure the efficiency and accuracy of defect detection.

The contributions of this work can be briefly summarized as follows:

- We propose the spatial pyramid feature fusion YOLOv5 (SPFFY), which adopts spatial pyramid dilated convolutions (SPDCs) to fuse the feature maps extracted in different receptive fields, thus integrating multi-scale defect information;

- The channel attention (CA) mechanism was utilized to evaluate the importance of the channels obtained from concatenate functions, which improves the representation ability of the model and generates more effective features;
- The sparsity training and pruning (STP) method based on the measurement of sparse and redundant features was utilized to obtain a smaller and more compact network while maintaining accuracy;
- The proposed method was evaluated on the PASCAL VOC and our AFP defect datasets, and based on the results, it performs better than the original models.

The remainder of this paper is organized as follows: Section 2 discusses related work. Section 3 describes the proposed methods in detail. Sections 4 and 5 present the experiments on the PASCAL VOC and AFP defect datasets, respectively. Finally, we conclude our work in Section 6.

## 2. Related Work

With the rapid development of deep learning in the field of object recognition, the AFP defect detection algorithm based on deep convolution neural networks (CNNs) has become a new research direction.

### 2.1. Deep CNNs for Object Detection

In recent years, deep convolution neural networks (CNNs) have achieved great success in visual recognition tasks [16–20]. With the improvement of hardware capability and the rapid development of deep convolutional neural network (CNN) architectures (AlexNet [16], VGGNet [21], ResNet [22], MobileNets [23,24], etc.), these models have powerful feature extraction capability to process large-scale images and are suitable for object recognition in complex scenes. The target recognition method based on CNNs is mainly divided into two-stage detection and one-stage detection [25]. Early two-stage recognition methods include R-CNN [26], SPP-Net [27], Fast R-CNN [28], and Faster R-CNN [29]. R-CNN and SPP-Net algorithms use SVM [30] for feature scoring and classification, which is complex to train and takes a long time to detect. Fast R-CNN uses the full connection layer instead of the SVM classifier, but it takes a long time to obtain the region of interest (ROI), and its detection speed is slow. Faster RCNN uses regional candidate networks (RPNs) to achieve end-to-end target recognition and detection, which improves the speed of target detection. However, as the two-stage target detection algorithm needs a large number of calculations and parameters, it cannot meet the requirements of real-time detection and batch application. One-stage detection methods include the YOLO series [31–35] and the SSD series [36,37]. When using the YOLO (You Only Look Once) algorithm for object recognition, the input image only needs one forward inference to predict all target positions and category information in the image. Each series of algorithms can further improve the recognition performance of the model by changing its classification strategy and backbone network.

### 2.2. Defect Detection in AFP

Various methods currently exist for AFP defect detection. In the AFP process, due to environmental factors, laying temperature, laying speed, laying pressure, equipment accuracy, laying trajectory planning, etc., different types of defects will occur in the final composite products. Many methods have been proposed based on machine vision to detect defects during the AFP process. Shadmehri et al. [38] proposed a laser vision detection system for the automatic fiber placement manufacturing process. This laser-assisted detection system is very intuitive but in essence is still based on manual detection, which does not significantly improve its efficiency. Marani et al. [39] used thermal imaging technology to obtain the surface image of glass fiber-reinforced materials. The SURF operator and unsupervised learning K-means are used to detect the surface defects of glass fiber composites. Denkena et al. [40] proposed a defect detection system based on infrared thermal imaging and related image processing for the inspection of AFP

processes. The edge detection algorithm is used to analyze the specific area compacted by the roller, extract the geometric shape and position of the tow, obtain the relevant information of the layer, and further detect defects such as overlaps, gaps, twists, etc. Brüning et al. [41] proposed a machine learning algorithm using an integrated infrared (IR) camera, which detects different types of defects and provides real-time quality information for the inspection of AFP processes, achieving automated data capture, data storage, modeling, and optimization. Chen et al. [42] proposed an intelligent AFP detection system that uses infrared vision for defect recognition and measurement and includes intelligent decision making, multi-parameter optimization, and data storage.

Some related studies in the field of deep learning have addressed AFP defect recognition [43]. Carsten Schmidt et al. [44] proposed a defect detection and classification method based on thermal imaging and deep learning in the automatic fiber placement (AFP) process. They designed three different CNN architectures for the detection and monitoring of tow defects, as well as for path monitoring. This method is only used to classify different defects and cannot locate them. In addition, when the defect target is small, the image contains a large amount of invalid background information, which interferes with the accuracy of classification. Sebastian Zambal et al. [8] proposed image segmentation to address defect detection in AFP and used artificially generated data [45] to solve the problem of insufficient defect data. The authors used probabilistic graphical models to generate training images and annotations and designed a neural network for image segmentation using an architecture similar to U-Nets, which is suitable for training with few real data. Sebastian Meister et al. [46] proposed a defect detection method based on convolutional and recurrent neural networks. In this method, one-dimensional signals are used to analyze the input height distribution of a laser line scanning sensor line by line, which is suitable for classifying images with large defects.

In these existing studies, the quality inspection of automatic fiber placement (AFP) is rarely addressed from the aspects of target defect recognition and the location of end-to-end learning and detection networks. Furthermore, the existing studies still cannot effectively solve the problem of background information interference in AFP defect detection. Thus, we aimed to design a deep learning algorithm to identify and analyze the defects of different scales and types in end-to-end frameworks and intuitively provide the inspection results.

### 2.3. Pruning

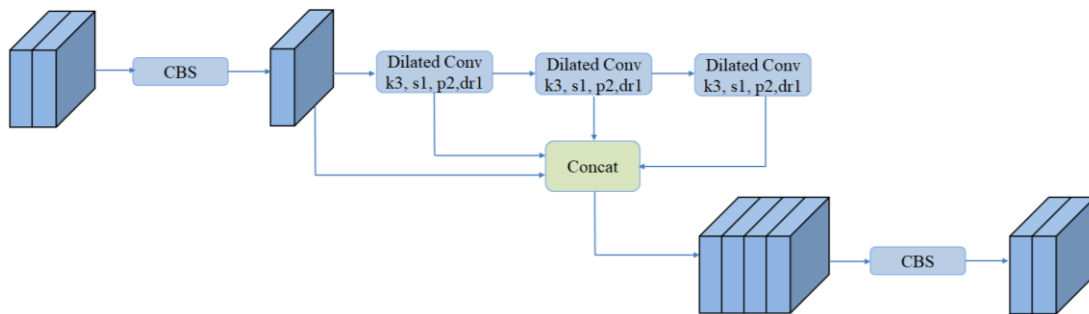
To achieve a more compact and effective network that eliminates the time-consuming detection of two-dimensional images, we utilized structured pruning for online AFP defect detection. Pruning methods commonly include unstructured and structured pruning. A pruning process consists of three steps: training large networks, pruning redundant channels, and retraining the pruned networks. Regarding unstructured pruning, LeCun et al. used second-derivative information and removed the weights based on their saliency [47]. The early weight pruning method is also mentioned in [48]. Han et al. [49,50] proposed a weight pruning framework to remove some CNN parameters and connections by pruning low-magnitude weights, thus achieving model compression. In contrast, structured pruning can be utilized to perform network slimming and computational acceleration, which do not require specialized hardware or libraries. Some studies [51–53] proposed a set of pruning criteria for CNNs to evaluate and remove unimportant feature channels and their corresponding kernels. In [54–56], sparsity regularization strategies were proposed to obtain sparse weights and features and reduce the time-intensiveness of the pruning–retraining step. In light of this body of research, we utilized feature sparsity training for the structured pruning and acceleration of CNNs to obtain a compact model.

## 3. Methods

In this section, the proposed method is described in detail. We present the architecture of our proposed method with the spatial pyramid feature fusion YOLOv5 (SPFFY), channel attention (CA), and sparsity training–pruning (STP).

### 3.1. Multi-Scale Feature Fusion

The original YOLOv5 utilizes a C3 architecture (a CSP bottleneck with 3 convolutions) with an SPPF (spatial pyramid pooling-fast) layer as the backbone to extract the feature map of the last convolution layer. The feature extraction capability of the backbone network directly affects the detection performance of ATP defects. Many recent studies [57,58] have revealed that the feature maps obtained from low-level convolutional layers have higher resolutions and, therefore, help to detect small objects. In these methods, a multi-scale spatial pyramid directs attention to the object by using its spatial features, which improves its detection. An SPPF block uses pooling layers with one-size kernels, and the output of each pooling becomes the input of the next pooling. Inspired by the SPPF, we propose a spatial pyramid dilated convolution (SPDC) module to fuse the multi-scale features extracted in different receptive fields in the same feature map, as shown in Figure 3. These modules replace SPPF and are further integrated with a channel attention mechanism. In the SPDC module, CBS represents conv + bn + silu. k3, s1, p2, and dr1 represent the convolution kernel, stride, padding, and dilation rate of size 3, 1, 2, and 1, respectively. SPDC modules can be regarded as a special block of CNN, as in these modules, the input and output feature maps have the same size; thus, they can be easily added to the backbone network of current detectors to obtain multi-scale feature maps. Here, we added an SPDC module behind each C3 module to replace the original SPPF in the backbone network of YOLOv5.



**Figure 3.** SPDC structure.

### 3.2. Channel Attention

In the existing network architectures, multi-scale features are obtained by concatenating the output features from different layers, but the importance of the output feature channel after concatenation is often ignored. In high-level layers, the extracted features often contain target feature information, and output channels have less redundant information. In low-level layers, by contrast, only simple edges and color blocks can be extracted, and the extracted features contain a large amount of background interference information. If the output feature channels extracted from high-level layers are directly concatenated with the low-level output features behind upsample, the target feature information undergoes interference. Therefore, we added a channel attention module after each concatenating operation in the neck part of the model, and the redundant feature channels can be ascribed to different weights for eliminating some noises.

The channel attention (CA) module assigns weights to fusion features from different scales. The channel attention mechanism is utilized after each concatenating operation in the neck network to direct more attention to the effective feature channels, as shown in Figure 4.

The CA module consists of two branches: multi-scale feature fusion and channel attention mechanism. The input feature maps after concatenation are represented as  $F_{in} \in \mathbb{R}^{H \times W \times C}$ . The feature fusion can generate the same sizes of output feature maps. The size of output feature maps with feature fusion is  $\widehat{F}_{in} \in \mathbb{R}^{H \times W \times C}$ . The channel attention mechanism contains two one-dimensional convolutional operations and the

sigmoid activation function, which can be used to obtain the weights of each channel. The  $i$ -th channel attention score is calculated as:

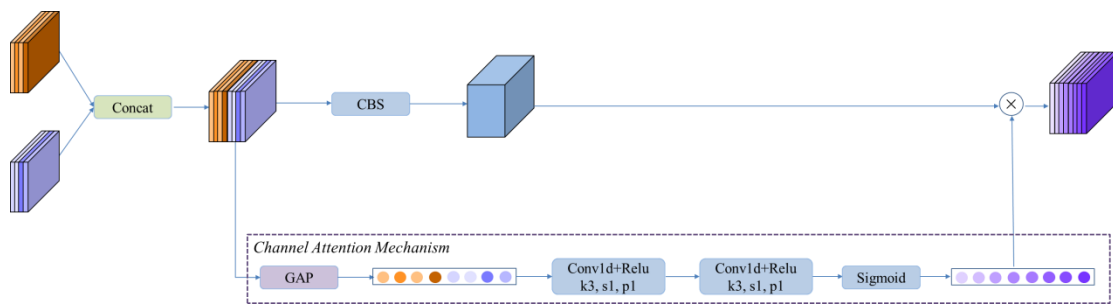
$$s_i = (1 + e^{-x_i})^{-1} \tag{1}$$

where the tensor  $x \in \mathbb{R}^{1 \times 1 \times C}$  is obtained from one-dimensional convolution operations.  $s \in \mathbb{R}^{1 \times 1 \times C}$  represents the weight of each feature channel. Then, the output feature channel is calculated as:

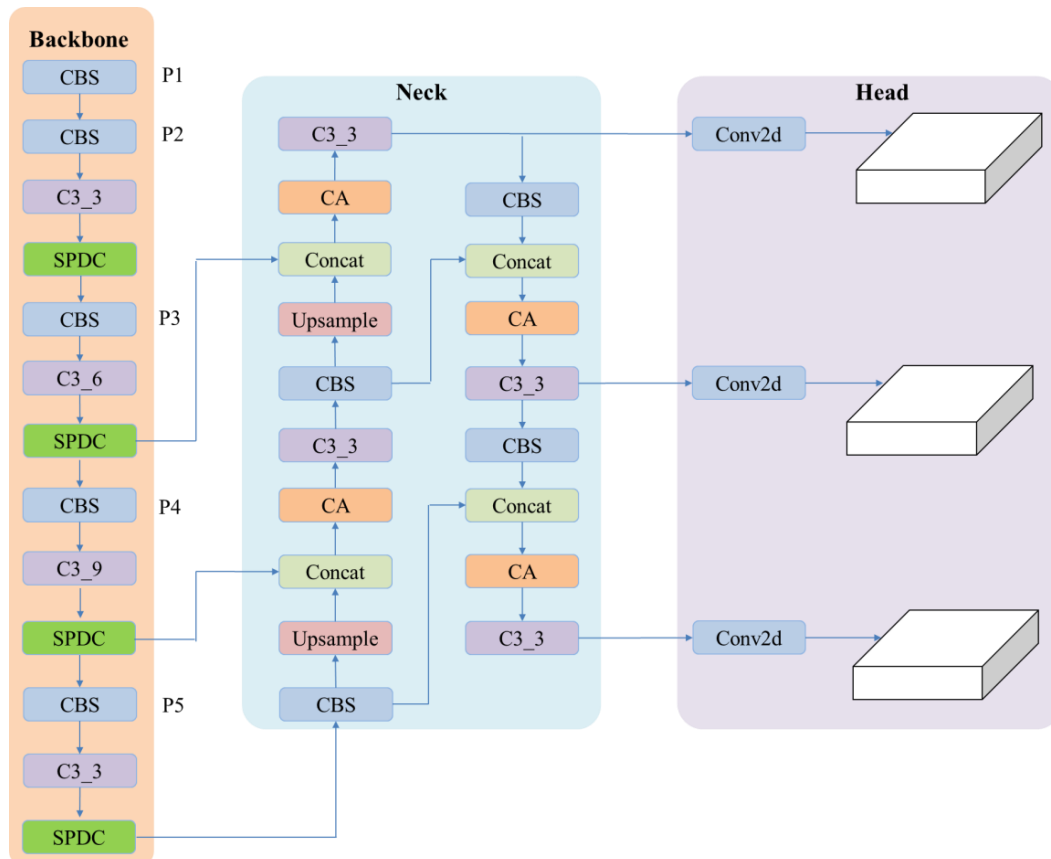
$$F_{Out} = s \times \widehat{F}_{in} \tag{2}$$

where the operation is performed by channel-wise multiplication between the score  $s$  and the feature map  $\widehat{F}_{in}$ .

The SPDC and CA modules are embedded in the backbone and neck network. The proposed model is illustrated in Figure 5.



**Figure 4.** The architecture of channel attention (CA) module. The cross (x) represents channel-wise multiplication.



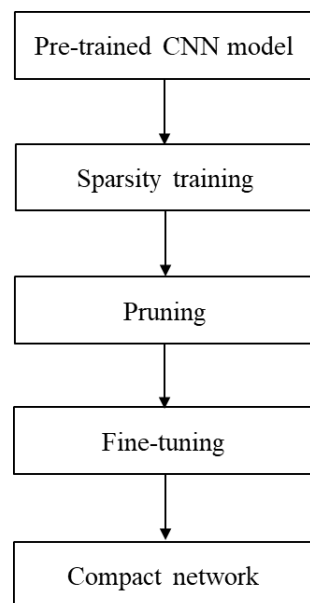
**Figure 5.** The architecture of the improved YOLOv5 with SPDC and CA modules.

### 3.3. Sparsity Training and Pruning

With the addition of SPDC and CA modules to the network, we introduced the sparsity training and pruning (STP) method to obtain more compact models and ensure the speed and accuracy of defect detection.

In general, the model compression rate can be determined by the actual use environment. However, when the compression rate is high, and the pretraining model has low sparsity, it is easy to prune the useful feature channels in the model, resulting in reduced detection accuracy. Therefore, in typical pruning methods, the number of channels to prune needs to be set to a small value in each iteration, and the pruning–retraining step needs to be repeated many times to obtain the final compact model. To avoid this, we employed the sparsity training of the pretrained networks to increase feature sparsity in each layer. We then used feature sparsity regularization on selected channels. During the sparsity training, the channels to be removed are penalized, and their outputs gradually decrease to zero. In this way, pruning can be finished in one iteration.

Different from most of the existing typical pruning methods [51,52], which adopt multiple iteration schemes (including pruning and retraining), our model needs only one iteration to perform sparsity training and pruning and achieve network slimming. The STP framework is illustrated in Figure 6. First, the location and number of convolutional kernels that need to be pruned are determined by calculating the sparse redundancy of each feature map. Then, sparsity constraint training is performed in each convolutional channel to be pruned, thus speeding up the sparsity of redundant channels and achieving one-step pruning and model precision recovery.



**Figure 6.** Sparsity training and pruning module.

The loss function is one of the important components of neural networks, which is used to calculate the gradients and update the weights of the network. The YOLOv5 loss function consists of three parts: class loss (BCE loss), objectness loss (BCE loss), and location loss (CIoU loss). It can be formulated as:

$$loss = \lambda_1 \mathcal{L}_{cls} + \lambda_2 \mathcal{L}_{obj} + \lambda_3 \mathcal{L}_{loc} \quad (3)$$

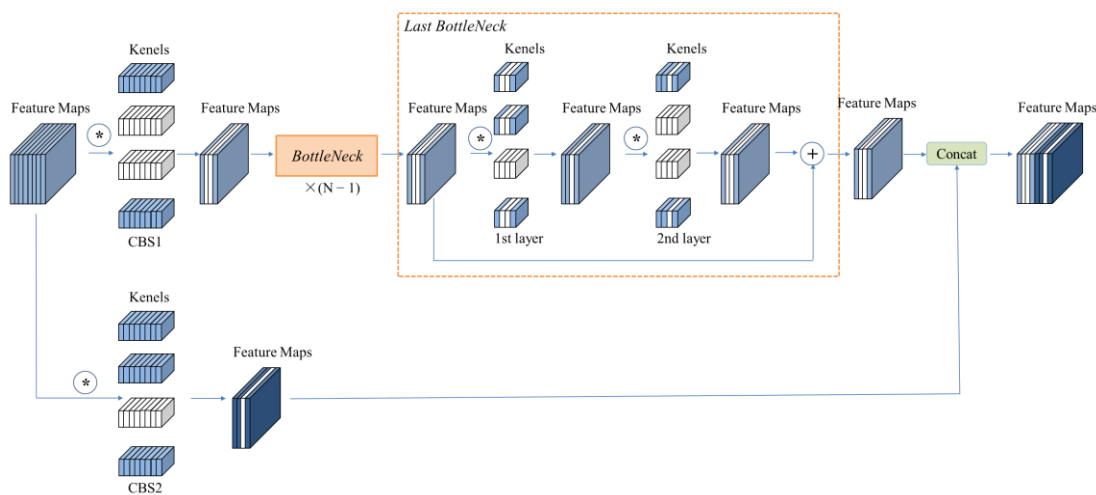
where  $\lambda_1, \lambda_2,$  and  $\lambda_3$  are the control parameters balancing these three terms. Additionally, the proposed loss function with sparsity training for CNNs is given by:

$$loss = \lambda_1 \mathcal{L}_{cls} + \lambda_2 \mathcal{L}_{obj} + \lambda_3 \mathcal{L}_{loc} + \lambda_p \sum_{l=1}^L R_p(F_l) \tag{4}$$

where  $R_p$  denotes the feature sparsity regularization on each layer and is calculated by the  $L_p$  norm of feature map  $F$ .

For the pruning process, different from a simple layer-stack structure, additional attention should be given to each special module of the proposed network.

For the backbone network, each block consists of C3 and SPDC modules. A C3 module with  $N$  bottlenecks is illustrated in Figure 7, where the symbol \* denotes the convolution operation, and the white blocks represent the pruned channels. The number of bottlenecks in the output channel needs to be consistent to finish the sum operation. We utilize the  $L_1$  norm of the feature map to evaluate the sparsity and redundancy of the output feature channels obtained by the element-wise addition of the last bottleneck in each C3 to determine the location and number of feature channels to be pruned. Then, the output channels of convolutional kernels corresponding to the second layer in each bottleneck are pruned. The importance of the output feature map of the first layer in each bottleneck is evaluated. Then, the corresponding output channels of kernels in the first layer and the input channels in the second layer can be pruned.



**Figure 7.** The architecture of C3-pruning module. \* denotes the convolution operation.

The pruning architecture of the SPDC module is shown in Figure 8, where the symbol \* denotes the convolution operation, and the white blocks represent the pruned channels. The importance of the feature maps obtained after the concatenation operation is first evaluated to determine the redundant feature channels. Then, the corresponding convolutional kernel channels of the previous layer and the input convolutional kernel channels of the next layer can be pruned.

For the channel attention (CA) module at the neck part of the proposed network, the CA-pruning module is illustrated in Figure 9, where the symbol \* denotes the convolution operation, and the white blocks represent the pruned channels.



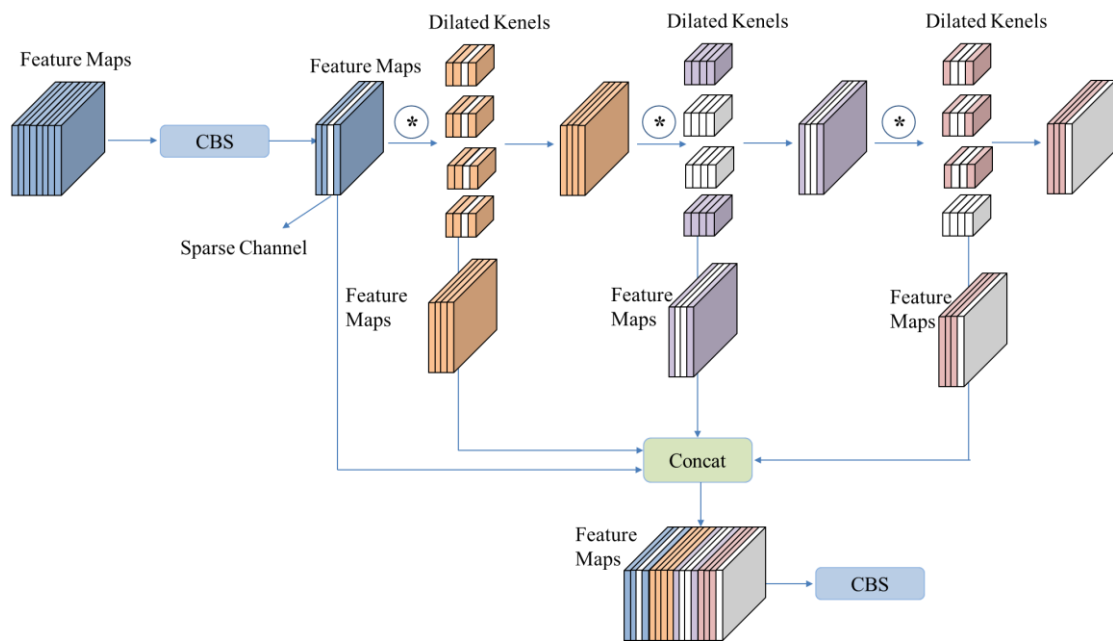


Figure 8. The architecture of SPDC-pruning module. \* denotes the convolution operation.

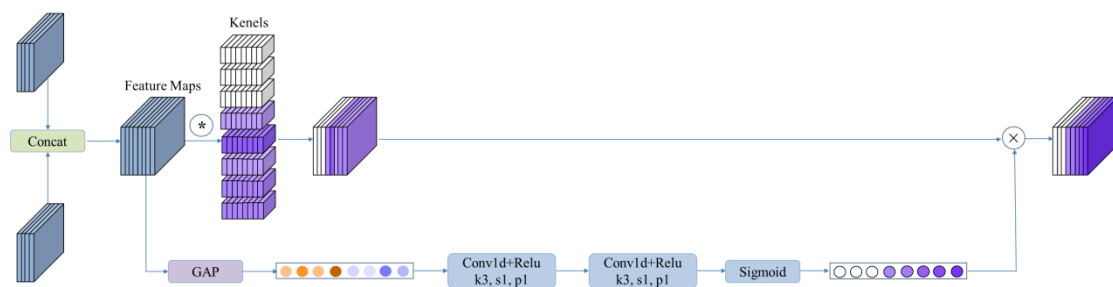


Figure 9. The architecture of CA-pruning module. The cross (x) represents channel-wise multiplication. \* denotes the convolution operation.

### 4. Experiments

In this section, we evaluate the effectiveness of the proposed SPFFY-CA on the benchmark PASCAL VOC dataset and our AFP defect dataset. Data augmentation methods, namely random crop, shifting, scaling, clipping, and random color jittering, were adopted to avoid overfitting. We trained the original network from scratch, defined as the baseline, using the computer with an Intel I7-8700 CPU and NVIDIA GTX 3060 with 12 GB of memory. YOLOv5 are open-source machine learning frameworks that accelerate the process from research prototyping to production deployment.

#### 4.1. Experiments on PASCAL VOC Datasets

The PASCAL Visual Object Classes Challenge (PASCAL VOC) dataset consists of VOC2007 and VOC2012. The dataset contains 20 objects, namely, Human: person; Animal: bird, cat, cow, dog, horse, and sheep; Vehicle: airplane, bicycle, boat, bus, car, motorbike, and train; indoor: bottle, chair, dining table, potted plant, sofa, and tv/monitor. The mean average precision (mAP) at the IoU threshold of 0.5 was calculated to measure the accuracy of target recognition. All the networks were trained on the datasets (16,551) containing VOC2007 and VOC2012 train-val datasets and were tested on the VOC2007 testing dataset (4952). In terms of the training details, the proposed models were trained using the SGD optimizer. The mini-batch size was 30, and an initial learning rate of  $10^{-2}$  was used. The momentum was 0.937, and the weight decay was 0.0005. The inference latency (batch size equal to 1) and parameters of the models were determined.

On the PASCAL VOC dataset, the performance of the proposed SPFFY-CA was compared with other state-of-the-art studies, and the results are shown in Table 1. It can be seen from Table 1 that SPFFY-CA-STP obtains 0.9% mAP higher than YOLOv5m with the same magnitude of parameters and latency time. Compared with other algorithms, SPFFY-CA and SPFFY-CA-STP have fewer parameters and higher recognition accuracy.

**Table 1.** Performance comparison of algorithms on VOC2007 test.

	Models	mAP	Params	Latency (ms)
	YOLOv2 [32]	76.8%	50.6 M	16
	YOLOv3 [34]	79.3%	62.1 M	28
	VGG16-SSD [36]	74.3%	26.2 M	29
	RefineDet [59]	80.0%	37.1 M	25
	FCOS [60]	77.8%	33.8 M	64
	ResNet101-CenterNet [61]	77.6%	52.2 M	23
	DLA34-CenterNet [61]	79.3%	21.2 M	20
	YOLOv5m [62]	89.9%	20.9 M	13
	YOLOv5l [62]	90.7%	46.5 M	18
Our Model	SPFFY-CA	<b>91.1%</b>	30.2 M	16
	SPFFY-CA-STP	<b>90.8%</b>	21.3 M	<b>13</b>

Table 2 shows the average precision (AP) of the proposed SPFFY-CA and SPFFY-CA-STP compared with SSD300 [36], SSD512 [36], CenterNet [61], and YOLOv5 [62]. It can be seen that the performance of the proposed method is superior to that of the other algorithms in the recognition performance of each category.

**Table 2.** Comparisons of all categories on the VOC2007 tests.

Method	YOLOv5m [62]	SSD300 [36]	SSD512 [36]	CenterNet [61]	SPFFY-CA-STP
mAP	89.9	74.3	76.8	79.3	<b>90.8</b>
tv	89.8	74.0	75.3	78.8	91.7
train	93.5	83.4	82.5	83.5	94.2
sofa	83.8	76.0	73.9	77.6	84.3
sheep	92.5	73.9	77.9	82.2	92.9
potted plant	69.5	48.6	50.3	55.6	70.1
person	94.1	76.2	79.7	83.6	95.1
motorbike	95.8	82.6	83.9	86.8	96.8
horse	95.8	85.3	85.2	87.5	96.6
dog	94.0	84.5	84.9	86.0	94.9
dining table	83.5	73.9	70.2	76.5	84.6
cow	94.0	78.3	83.1	84.0	94.7
chair	77.1	54.7	57.8	62.8	78.2
cat	94.5	86.1	86.0	85.8	95.6
car	95.7	84.2	87.5	87.0	96.7
bus	96.0	83.0	86.2	84.5	95.4
bottle	85.9	47.6	53.2	64.0	87.8
boat	81.2	66.3	73.8	69.8	81.9
bird	90.8	72.3	78.4	78.2	92.0
bicycle	94.1	80.2	84.7	86.8	95.6
airplane	96.7	75.5	82.4	85.6	97.1

#### 4.2. Ablation Study

We conducted ablation studies to validate the proposed method as follows:

Spatial pyramid dilated convolutions (SPDC): We investigated the power of the spatial pyramid dilated convolution module by comparing the SPFFY-CA with and without the SPDC module. For this experiment, we used the SPFFY-CA without SPDC and trained it on the PASCAL VOC dataset. The training strategy was the same as in the previous section. The performance comparison results are shown in Table 3. It can be seen that the SPFFY-CA with the SPDC module can obtain better performance.

**Table 3.** Performance comparison of SPFFY-CA with and without SPDC.

Model	Test Error	Parameters
SPFFY-CA (baseline)	91.1%	30.2 M
SPFFY-CA without SPDC	90.4%	22.9 M

Channel attention (CA): In this experiment, we studied the effects of SPFFY-CA with and without the multi-scale channel attention (CA) module. We used the SPFFY-CA without the CA module and trained the model on the PASCAL VOC dataset. The training strategy was the same as in the previous experiment. The performance comparison results are shown in Table 4. It can be seen that the SPFFY-CA with the CA module can obtain better performance.

**Table 4.** Performance comparison of SPFFY-CA with and without CA.

Model	Test Error	Parameters
SPFFY-CA (Baseline)	91.1%	30.2 M
SPFFY-CA without CA	90.7%	28.2 M

Sparsity Training and Pruning (STP): In this experiment, we investigated the effect of sparsity training and pruning (STP) on SPFFY-CA. We used the SPFFY-CA trained on the PASCAL VOC dataset. The training strategy was the same as in the previous experiment. The SPFFY-CA model was pruned with three different compression rates, and the results are shown in Table 5, where “SPFFY-CA-pruned-2” is based on the model of “SPFFY-CA-pruned-1”. From Table 5, it can be inferred that the STP can compress the SPFFY-CA model and ensure the stability of identification accuracy.

**Table 5.** Performance comparison of SPFFY-CA three different compression rates.

Model	Test Error	Parameters
SPFFY-CA (baseline)	91.1%	30.2 M
SPFFY-CA-pruned-1	90.9%	24.6 M
SPFFY-CA-pruned-2	90.8%	21.3 M

## 5. Experiments on AFP Defect Datasets

Due to the complexity of the AFP manufacturing process, as well as environmental factors, process parameters, CFRP defects, equipment accuracy, laying trajectory planning, etc., different types of defects will appear in the final composite products, which will affect their mechanical properties [63,64]. Common types of AFP defects include wrinkles, twists, gaps, bubbles, and the presence of foreign material. In this study, an AFP defect dataset was labeled with 3000 images with an original resolution of  $1000 \times 1000$ . Then, 80% of the defect samples were used as the train-val dataset, and the rest were used as the test set to evaluate the performance of the model. The number of instances found for each type of defect is shown in Figure 10.

The mean average precision (mAP) at the IoU threshold of 0.5 was calculated to measure the accuracy of defect recognition. In terms of the training details, the models were trained using the SGD optimizer with a batch size of 30 and an initial learning rate of  $10^{-2}$ . The momentum was 0.9, and the weight decay was 0.0005. The latency time (batch size equal to 1) was determined. On the AFP defect dataset, the performance of the proposed SPFFY-CA was compared with other detection algorithms, and the results are shown in Table 6. It can be seen from Table 6 that the SPFFY-CA proposed in this paper achieves an accuracy of 93.1% on the AFP defect dataset and has higher recognition confidence than YOLOv5m for defect detection. For all the various types of defects, SPFFY-CA achieves higher performance than YOLOv5m.

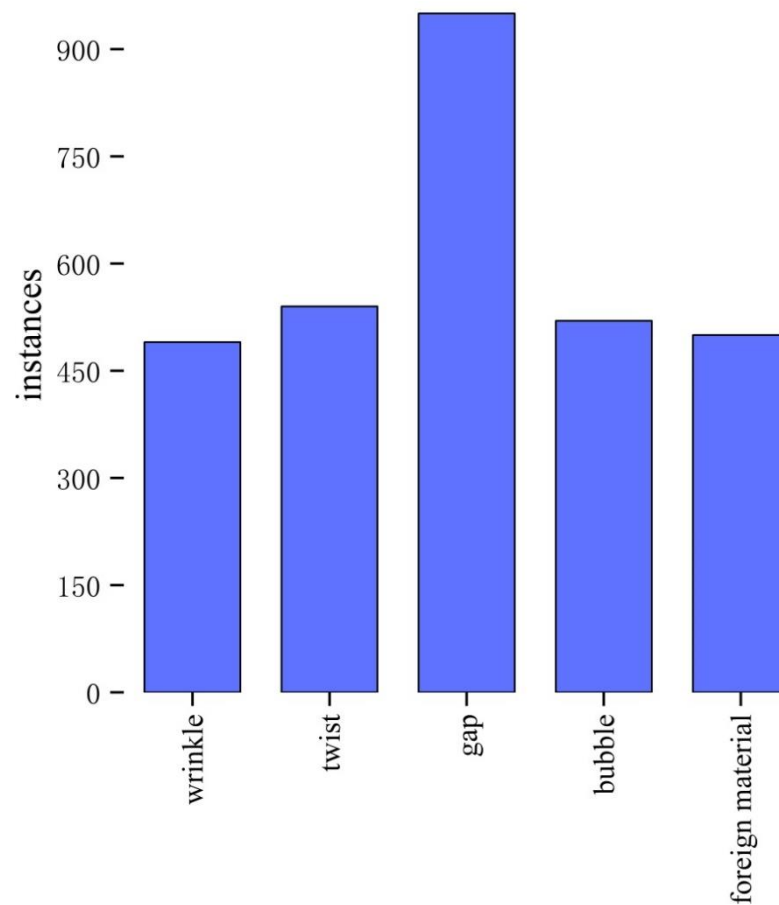


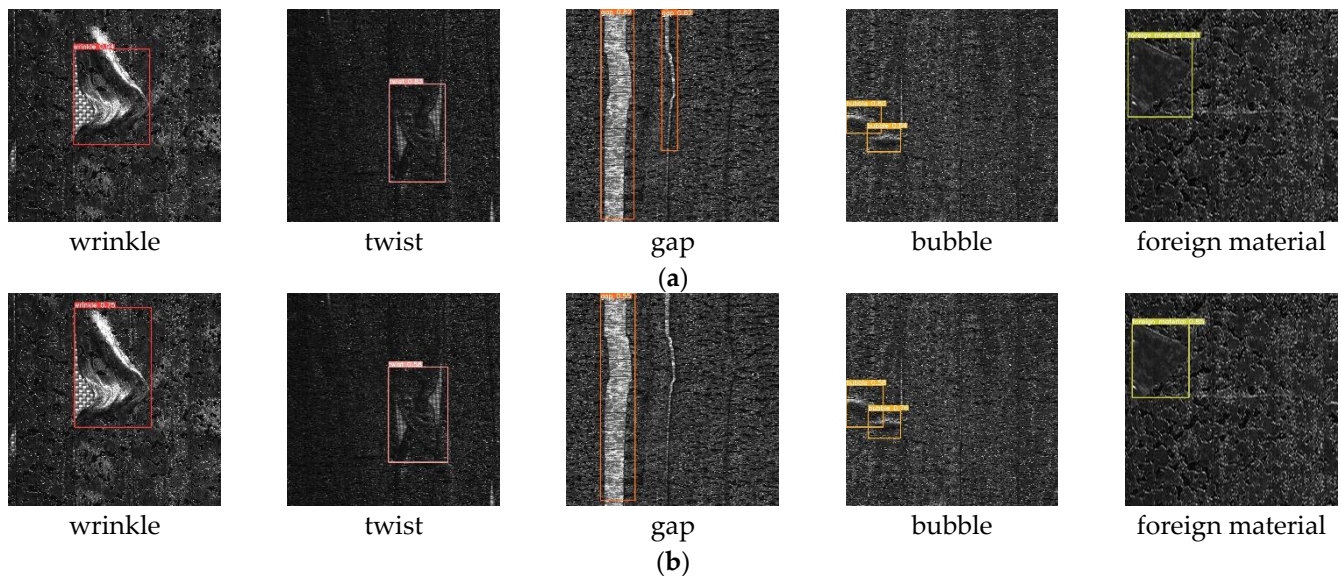
Figure 10. The number of instances for each type of defect.

Table 6. Performance comparison of algorithms on the AFP defect dataset.

Models	mAP (%)	AP (%)				
		Wrinkle	Twist	Gaps	Bubble	Foreign Material
YOLOv5m (Baseline)	92.0	96.1	90.5	83.2	92.1	97.9
SPFFY-CA-STP	<b>93.1</b>	97.2	91.2	85.6	93.2	98.3

Figure 11 shows the detection results of SPFFY-CA-STP and the original YOLOv5m for various defects, where the confidence score is higher than 0.5. Figure 11a shows the recognition effect of the designed SPFFY-CA-STP model, and Figure 11b shows the detection effect of the original YOLOv5m. It can be seen from Figure 11 that the SPFFY-CA-STP model has higher recognition confidence than YOLOv5m for defects of different scales and types.

The proposed SPFFY-CA-STP can achieve higher performance on the PASCAL VOC dataset while maintaining the same detection speed and can realize the real-time detection of multi-scale AFP defects. The quality inspection of automatic fiber placement (AFP) is thus addressed through the recognition of target defects and the location of an end-to-end learning and detection network, but the main limitation is that this method is mainly used when the defect types are known, and further data collection is required for unknown defects.



**Figure 11.** The detection results of SPFFY-CA-STP and original YOLOv5m for various defects: (a) detection results of the proposed SPFFY-CA-STP; (b) detection results of YOLOv5m.

## 6. Conclusions and Future Work

In this paper, we proposed a multi-scale AFP defect detection algorithm named the spatial pyramid feature fusion YOLOv5 with channel attention (SPFFY-CA), which includes spatial pyramid dilated convolutions (SPDCs) and channel attention (CA) modules to fuse the feature maps extracted in different receptive fields, thus integrating multi-scale defect information. Through the CA mechanism, the importance of the channels obtained from the concatenate function was evaluated, and further attention was given to the effective feature channels, which improved the representation ability and generated more effective features. In addition, we employed the sparsity training and pruning (STP) method to obtain more compact models and ensure the speed and accuracy of defect detection. The experimental results on the PASCAL VOC and the AFP defect datasets prove the effectiveness of the proposed approach and that it can obtain state-of-the-art performance. In future research, we will further study the defect detection of manual paving and better apply the visual identification technology to control the composite manufacturing process.

**Author Contributions:** Conceptualization, Y.Z. and W.W.; methodology, Y.Z.; software, W.W.; validation, W.W., Q.L., Z.G. and Y.J.; formal analysis, Y.Z. and W.W.; investigation, Y.Z. and W.W.; resources, Y.Z.; data curation, W.W.; writing—original draft preparation, Y.Z. and W.W.; writing—review and editing, Y.Z., W.W., Q.L., Z.G. and Y.J.; visualization, Y.Z. and W.W.; supervision, W.W.; project administration, Y.Z.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

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

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author (wangwei\_4524@163.com). The data are not publicly available due to privacy restrictions.

**Acknowledgments:** We sincerely thank the anonymous reviewers for their critical comments and suggestions for improving the manuscript.

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

## References

1. Soutis, C. Fibre reinforced composites in aircraft construction. *Prog. Aerosp. Sci.* **2005**, *41*, 143–151. [[CrossRef](#)]
2. Avdelidis, N.; Almond, D.; Dobbison, A.; Hawtin, B.; Castanedo, C.I.; Maldague, X. Aircraft composites assessment by means of transient thermal NDT. *Prog. Aerosp. Sci.* **2004**, *40*, 143–162. [[CrossRef](#)]

3. Denkena, B.; Schmidt, C.; Weber, P. Automated fiber placement head for manufacturing of innovative aerospace stiffening structures. *Procedia Manuf.* **2016**, *6*, 96–104. [[CrossRef](#)]
4. Kozaczuk, K. Automated fiber placement systems overview. *Pr. Inst. Lotnictwa* **2016**, *245*, 52–59. [[CrossRef](#)]
5. Belhaj, M.; Hojjati, M. Wrinkle formation during steering in automated fiber placement: Modeling and experimental verification. *J. Reinf. Plast. Compos.* **2018**, *37*, 396–409. [[CrossRef](#)]
6. August, Z.; Ostrander, G.; Michasiow, J.; Hauber, D. Recent developments in automated fiber placement of thermoplastic composites. *SAMPE J.* **2014**, *50*, 30–37.
7. Harik, R.; Saidy, C.; Williams, S.J.; Gurdal, Z.; Grimsley, B. Automated Fiber Placement Defect Identity Cards: Cause, Anticipation, Existence, Significance, and Progression. 2018. Available online: <https://ntrs.nasa.gov/api/citations/20200002536/downloads/20200002536.pdf> (accessed on 8 October 2022).
8. Zambal, S.; Heindl, C.; Eitzinger, C.; Josef, S. End-to-end defect detection in automated fiber placement based on artificially generated data, Fourteenth international conference on quality control by artificial vision. *SPIE* **2019**, *11172*, 371–378.
9. Arns, J.Y.; Oromiehie, E.; Arns, C.; Gangadhara, P.B. Micro-CT analysis of process-induced defects in composite laminates using AFP. *Mater. Manuf. Process.* **2021**, *36*, 1561–1570. [[CrossRef](#)]
10. Nguyen, M.H.; Vijayachandran, A.A.; Davidson, P.; Call, D.; Lee, D.; Waas, A.M. Effect of automated fiber placement (AFP) manufacturing signature on mechanical performance of composite structures. *Compos. Struct.* **2019**, *228*, 111335. [[CrossRef](#)]
11. 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.* **2014**, *27*, 431–445. [[CrossRef](#)]
12. Sacco, C.; Radwan, A.B.; Harik, R.; Tooren, M.V. Automated fiber placement defects: Automated inspection and characterization. In Proceedings of the SAMPE 2018 Conference and Exhibition, Long Beach, CA, USA, 21–24 May 2018. No. NF1676L-29116.
13. Meister, S.; Wermes, M.; Stüve, J.; Groves, R.M. Investigations on Explainable Artificial Intelligence methods for the deep learning classification of fibre layup defect in the automated composite manufacturing. *Compos. Part B Eng.* **2021**, *224*, 109160. [[CrossRef](#)]
14. Tang, Y.; Wang, Q.; Cheng, L.; Li, J.; Ke, Y. An in-process inspection method integrating deep learning and classical algorithm for automated fiber placement. *Compos. Struct.* **2022**, *300*, 116051. [[CrossRef](#)]
15. Meister, S. Automated Defect Analysis Using Optical Sensing and Explainable Artificial Intelligence for Fibre Layup Processes in Composite Manufacturing. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 2022.
16. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *1*, 1097–1105.
17. Kasprzak, W.; Jankowski, B. Light-Weight Classification of Human Actions in Video with Skeleton-Based Features. *Electronics* **2022**, *11*, 2145. [[CrossRef](#)]
18. Gowda, K.M.V.; Madhavan, S.; Rinaldi, S.; Divakarachari, P.B.; Atmakur, A. FPGA-Based Reconfigurable Convolutional Neural Network Accelerator Using Sparse and Convolutional Optimization. *Electronics* **2022**, *11*, 1653. [[CrossRef](#)]
19. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [[CrossRef](#)]
20. Abunadi, I.; Senan, E.M. Deep Learning and Machine Learning Techniques of Diagnosis Dermoscopy Images for Early Detection of Skin Diseases. *Electronics* **2021**, *10*, 3158. [[CrossRef](#)]
21. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
22. 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, Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
23. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
24. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L. Mobilenetv2, Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
25. Du, L.; Zhang, R.; Wang, X. Overview of two-stage object detection algorithms. *J. Phys. Conf. Ser. IOP Publ.* **2020**, *1544*, 012033. [[CrossRef](#)]
26. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [[CrossRef](#)] [[PubMed](#)]
28. Girshick, R. Fast r-cnn. In Proceedings of the IEEE international conference on computer vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
29. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [[CrossRef](#)] [[PubMed](#)]
30. Jakkula, V. Tutorial on support vector machine (svm). *Sch. EECS Wash. State Univ.* **2006**, *37*, 3.
31. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
32. Redmon, J.; Farhadi, A. YOLO9000, Better, faster, stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; 2017; pp. 7263–7271.

33. Huang, R.; Pedoeem, J.; Chen, C. YOLO-LITE: A real-time object detection algorithm optimized for non-GPU computers. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–14 December 2018; pp. 2503–2510.
34. Redmon, J.; Farhadi, A. Yolov3, An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
35. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4, Optimal speed and accuracy of object detection. *arxiv* **2020**, arXiv:2004.10934.
36. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. *Ssd: Single Shot Multibox Detector*. *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
37. Jeong, J.; Park, H.; Kwak, N. Enhancement of SSD by concatenating feature maps for object detection. *arXiv* **2017**, arXiv:1705.09587.
38. Shadmehri, F.; Ioachim, O.; Pahud, O.; Brunel1, J.; Landry, A.; Hoa, S.V.; Hojjati, M. Laser-vision inspection system for automated fiber placement (AFP) process. In Proceedings of the 20th International conference on composite materials Copenhagen, Copenhagen, Denmark, 19–24 July 2015.
39. Marani, R.; Palumbo, D.; Galietti, U.; Stella, E.; D’Orazio, T. Automatic detection of subsurfacedefects in composite materials using thermography and unsupervised machine learning. In Proceedings of the IEEE International Conference on Intelligent Systems, Sofia, Bulgaria, 4–6 September 2016.
40. Denkena, B.; Schmidt, C.; Völtzer, K.; Hocke, T. Thermographic online monitoring system for AutomatedFiber Placement processes. *Compos. Part B Eng.* **2016**, *97*, 239–243. [[CrossRef](#)]
41. Brüning, J.; Denkena, B.; Dittrich, M.A.; Hocke, T. Machine learning approach for optimization of automated fiber placement processes. *Procedia CIRP* **2017**, *66*, 74–78. [[CrossRef](#)]
42. Chen, M.; Jiang, M.; Liu, X.; Wu, B. Intelligent Inspection System Based on Infrared Vision for Automated Fiber Placement. In Proceedings of the 2018 IEEE International Conference on Mechatronics and Automation (ICMA), Changchun, China, 5–8 August 2018; pp. 918–923.
43. Meister, S.; Wermes, M.A.M.; Stüve, J.; Groves, R.M. Review of image segmentation techniques for layup defect detection in the Automated Fiber Placement process. *J. Intell. Manuf.* **2021**, *32*, 2099–2119. [[CrossRef](#)]
44. Schmidt, C.; Hocke, T.; Denkena, B. Deep learning-based classification of production defects in automated-fiber-placement processes. *Prod. Eng.* **2019**, *13*, 501–509. [[CrossRef](#)]
45. Mueller, F.; Bernard, F.; Sotnychenko, O.; Mehta, D.; Sridhar, S.; Casas, D.; Theobalt, C. GANerated hands for real-time 3d hand tracking from monocular RGB. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
46. Meister, S.; Wermes, M. Performance evaluation of CNN and R-CNN based line by line analysis algorithms for fibre placement defect classification. *Prod. Eng.* **2022**, 1–16. [[CrossRef](#)]
47. LeCun, Y.; Denker, J.S.; Solla, S.A. Optimal Brain Damage. Available online: <https://proceedings.neurips.cc/paper/1989/file/6c9882bbac1c7093bd25041881277658-Paper.pdf> (accessed on 8 October 2022).
48. Hassibi, B.; Stork, D.G. Second order derivatives for network pruning: Optimal Brain 495 Surgeon. In *Advances in Neural Information Processing Systems*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1992; pp. 164–171.
49. Han, S.; Mao, H.; Dally, W.J. Deep compression: Compressing deep neural networks 497 with pruning, trained quantization and huffman coding. *arXiv* **2015**, arXiv:1510.00149.
50. Han, S.; Pool, J.; Tran, J.; Dally, W. Learning both weights and connections for efficient 500 neural network. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1135–1143.
51. Li, H.; Kadav, A.; Durdanovic, I.; Samet, H.; Graf, H.P. Pruning filters for efficient convnets, in: International Conference on Learning Representations (ICLR). *arxiv* **2017**, arXiv:1608.08710.
52. Hu, H.; Peng, R.; Tai, Y.-W.; Tang, C.-K. Network Trimming: A data-driven neuronpruning approach towards efficient deep architectures. *arXiv* **2016**, arXiv:1607.03250.
53. Wang, W.; Zhu, L.; Guo, B. Reliable identification of redundant kernels for convolutional neural network compression. *J. Vis. Commun. Image Represent* **2019**, *63*, 102582. [[CrossRef](#)]
54. Wen, W.; Wu, C.; Wang, Y.; Chen, Y.; Li, H. Learning structured sparsity in deep neural networks. Proceedings of 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, 5–10 December 2016; pp. 2074–2082.
55. Liu, Z.; Li, J.; Shen, Z.; Huang, Y.; Zhang, C. Learning efficient convolutional networks through network slimming. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2755–2763.
56. Wang, W.; Zhu, L. Structured feature sparsity training for convolutional neural network compression. *J. Vis. Commun. Image Represent.* **2020**, *71*, 102867. [[CrossRef](#)]
57. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE international conference on computer vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
58. Lin, T.Y.; Dollár, 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, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
59. Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; Li, S.Z. Single-shot refinement neural network for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2018), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4203–4212.
60. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: Fully convolutional one-stage object detection. In Proceedings of the 2019 IEEE/CVF international conference on computer vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 9627–9636.
61. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as points. *arxiv* **2019**, arXiv:1904.07850.

62. Ultralytics. Yolov5. Available online: <https://github.com/ultralytics/yolov5> (accessed on 8 October 2022).
63. Lozano, G.G.; Tiwari, A.; Turner, C.; Astwood, S. A review on design for manufacture of variable stiffness composite laminates. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* **2015**, *230*, 981–992. [[CrossRef](#)]
64. Abouhamzeh, M.; Nardi, D.; Leonard, R.; Sinke, J. Effect of prepreg gaps and overlaps on mechanical properties of fibre metal laminates. *Compos. Part A Appl. Sci. Manuf.* **2018**, *114*, 258–268. [[CrossRef](#)]