

Article

Lightweight CNN-Based Method for Spacecraft Component Detection

Yuepeng Liu ¹, Xingyu Zhou ²  and Hongwei Han ^{2,3,*}¹ School of Mechanical and Electrical, Beijing Institute of Technology, Beijing 100081, China² School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China³ Beijing Institute of Technology Chongqing Innovation Center, Chongqing 401120, China

* Correspondence: hanhongweibit@163.com

Abstract: Spacecraft component detection is essential for space missions, such as for rendezvous and on-orbit assembly. Traditional intelligent detection algorithms suffer from drawbacks related to high computational burden, and are not applicable for on-board use. This paper proposes a convolutional neural network (CNN)-based lightweight algorithm for spacecraft component detection. A lightweight approach based on the Ghost module and channel compression is first presented to decrease the amount of processing and data storage required by the detection algorithm. To improve feature extraction, we analyze the characteristics of spacecraft imagery, and multi-head self-attention is used. In addition, a weighted bidirectional feature pyramid network is incorporated into the algorithm to increase precision. Numerical simulations show that the proposed method can drastically reduce the computational overhead while still guaranteeing good detection precision.

Keywords: spacecraft component detection intelligent algorithm; lightweight



Citation: Liu, Y.; Zhou, X.; Han, H. Lightweight CNN-Based Method for Spacecraft Component Detection. *Aerospace* **2022**, *9*, 761. <https://doi.org/10.3390/aerospace9120761>

Academic Editor: Fabio Ferrari

Received: 2 November 2022

Accepted: 24 November 2022

Published: 27 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

On-orbit service has gradually become a research hotspot in the aerospace field [1,2]. The dynamics and control of spacecraft and space situational awareness are the basis of on-orbit service [3–5], in which spacecraft detection is an important technology for providing necessary information for subsequent space missions [6–8]. The detection of spacecraft components is the first step of any on-orbit service, such as on-orbit refueling, on-orbit maintenance of spacecraft, and classical on-orbit inspection of a spacecraft formation [9]. It is necessary to detect the components of the spacecraft and then locate the major components. Spacecraft component detection can accurately detect the position and type of spacecraft components. The main components of spacecraft include the main body, solar panels, antennae, etc. [10].

The current methods for spacecraft component detection can be broadly divided into two kinds, namely nonintelligent and intelligent methods. Nonintelligent spacecraft component detection is mainly based on the data characteristics of the target to obtain target models or manually set features. In [11], a kind of illumination fuzzy similarity fusion invariant moment was proposed to effectively solve the problem of spacecraft target detection with different poses and illumination conditions. In [12], a target detection method based on the improved directional gradient histogram feature was proposed. Reference [13] proposed a multiple-feature fusion spacecraft local detection method. Good detection results were obtained, although the generalization ability of the model was not good. Nonintelligent spacecraft component detection methods have high complexity and require more running time and storage space. They have poor generalization ability and do not consider the texture interference caused by the Earth's background, which makes it difficult for them to meet the challenges of many uncertain factors in spacecraft component detection.

With the development of artificial intelligence, object detection technology has made great breakthroughs and can meet the mission requirements in terms of accuracy and real-time performance. Due to the particularity of spacecraft missions, spacecraft component detection based on deep learning is still in its infancy, and there are few related research works. Reference [14] used the YOLO model to identify spacecraft components under different perspectives, distances, and occlusions. The results show that the accuracy of the model was more than 90%. Reference [15] applied Mask R-CNN (Mask region-based convolutional neural network) to achieve space spacecraft feature detection. Taking the R-FCN (region-based fully convolutional network) and light-head R-CNN as references, the model was optimized and improved to improve the detection efficiency. Reference [16] proposed an improved multi-layer convolutional neural network based on LeNet for spatial object recognition, which improved recognition accuracy. Reference [17] proposed an R-CNN spacecraft component detection algorithm based on regional convolutional neural networks. On the basis of Mask R-CNN, a new feature extraction structure was constructed by combining DenseNet, ResNet, and FPN. The feature propagation between layers was enhanced by the idea of dense connection. Because of the poor set of training samples, the software was used to construct spacecraft images with different angles and heights.

However, the number of model parameters of the R-CNN and LeNet algorithms is too large to run on space-borne equipment with limited computing power. Moreover, many existing methods [13–16] fail to fully consider the interference of illumination, noise, and the Earth background.

This paper proposes a lightweight spacecraft component detection algorithm based on the Ghost module, namely YOLOv5-GMB. Taking YOLOv5 [18] as the framework, the algorithm is compressed by combining the improved Ghost module and the channel compression method. To overcome the degradation of spacecraft component detection performance, this paper introduces multi-head self-attention (MHSA) [19], which uses the characteristics of MHSA's global attention to improve the ability to capture different information. Meanwhile, data with different scales are fused by a weighted bidirectional feature pyramid network (BiFPN) [20]. The effectiveness of the proposed method is improved, and the robustness of the algorithm is enhanced. We expand the dataset by enhancing data augmentation to make the model more robust to images acquired in different environments. At the same time, the detection ability of objects with different sizes is improved.

The remaining paper is organized as follows. Section 2 introduces the algorithm flow and the construction of the spacecraft dataset. Section 3 introduces lightweight methods. Section 4 gives the details of the experiment and the test results, and then analyses the test results. Finally, Section 5 contains the conclusion.

2. Description of Spacecraft Component Detection

2.1. Component Detection Process

As the first step of on-orbit service, spacecraft component detection provides the type and location information of spacecraft components for subsequent missions. Due to the motion of spacecraft and the rotation of the Earth, the brightness, shape, and Earth background are constantly changing, which brings challenges for detection. Considering these factors, to achieve high accuracy and lightweight spacecraft component detection, this paper proposes a CNN-based spacecraft component detection model named YOLOv5-GMB. The overall flow of the algorithm is shown in Figure 1. The steps are as follows:

Step 1. Input the image to be detected;

Step 2. Image features are extracted using the CSPDarknet53 fusion Ghost module, and then spatial pyramid pooling (SPP) is used to convert the feature maps of different sizes into fixed-size feature vectors to achieve fusion with different features;

Step 3. Implement global self-attention using MHSA to improve the ability to capture global information;

Step 4. Here, BiFPN is used to introduce learnable weights to learn the importance of different input features. At the same time, top-down and bottom-up multiscale feature fusion are repeatedly applied to stitch feature maps of different scales;

Step 5. Input the feature map into YOLO Head for object classification. Finally, the classification results and detection frames are generated. Then, the classification results and position information of each component of the spacecraft are obtained;

Step 6. Output the detection images and spacecraft component information.

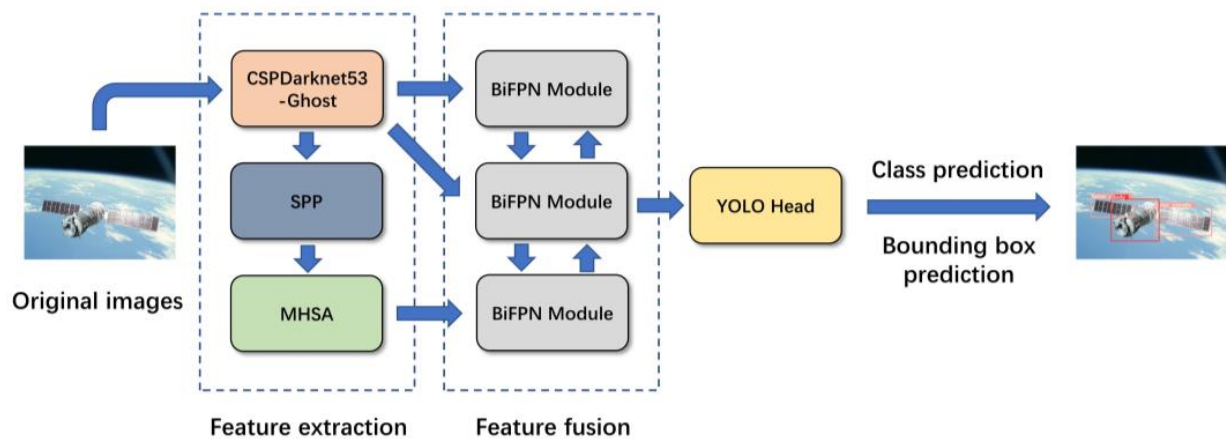


Figure 1. Overall flowchart of component detection.

2.2. Dataset Construction

Remote sensing image and spacecraft attitude estimate datasets, such as RSSCN7 and SPEED, make up the majority of the space field's current datasets. There are few corresponding datasets for spacecraft component detection due to the difficulty of image capture and the significant background interference. We gathered 1000 spacecraft photos from fake or real images and movies from space organizations that had been issued concerning spacecraft. Some spacecraft photographs are combined with the view of Earth or other planets using image fusion technology. A dataset of 1000 spaceship photos is created by reducing the resolution of select photographs to account for the onboard camera's low pixel count. The dataset is separated into training and test sets with a ratio of 80% to 20% (800 and 200, respectively). The fully built spacecraft dataset is shown in Figure 2 through a number of samples.



Figure 2. Several samples in the built spacecraft dataset.

3. Lightweight Algorithm for Spacecraft Component Detection

3.1. Baseline Detection Algorithm

The two categories of currently-used lightweight techniques are model compression and compact model design. Pruning, model quantization, and knowledge distillation are the key techniques used in model compression, which is applied to large existing models. A number of small models, including MobilenetV2-V3 [21,22], ShufflenetV2 [23], and GhostNet [24], have been presented in response to the necessity of deploying neural networks on embedded devices. These models use a small number of FLOPs to attain good performance. Although model compression on currently large models makes the model simpler, the smaller model is still quite large. It is feasible to create an algorithm that can function on hardware with low processing power by using a compact model. In the ImageNet classification test, three lightweight CNN algorithms are evaluated. The experimental data are displayed in Figure 3. The author of MobilenetV3 only provides two versions, small and large. While the other algorithms provide three versions, we test all versions. On ImageNet validation datasets, all results are presented with single-class top-1 accuracy. When the model FLOPs are the same, the findings show that GhostNet has the highest detection accuracy. Therefore, to create lightweight algorithms, we use the Ghost module in the optimal GhostNet model and channel compression.

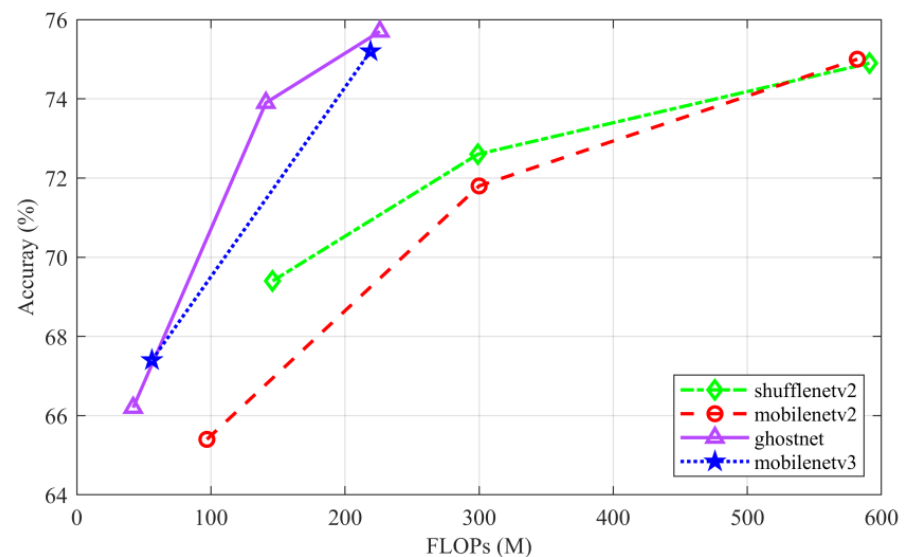


Figure 3. Performance comparison of compact models.

3.2. Lightweight of Network Structure

The one-stage YOLO series, which has distinct benefits in real-time detection and precision, is crucial for the task of object detection. As a result, we pick YOLOv5 as the starting point. We utilize YOLOv5s as the framework, since being lightweight is important. The YOLO head serves as the detector, the BiFPN serves as the algorithm's neck, and the MHSA fused by CSPMarknet53 serves as the algorithm's backbone. Because of the Ghost module and channel compression, the approach is lightweight. Figure 4 depicts the structure of the algorithm.

With fewer arguments, the Ghost module creates more features. In comparison to the standard CNN, the Ghost module's computational cost and overall parameter requirements are lower, but the size of the output feature map remains constant. Theoretical speedup using the Ghost module is as follows:

$$\begin{aligned}
 r_s &= \frac{n \cdot h' \cdot w' \cdot c \cdot k \cdot k}{\frac{n}{s} \cdot h' \cdot w' \cdot c \cdot k \cdot k + (s-1) \cdot \frac{n}{s} \cdot h' \cdot w' \cdot d \cdot d} \\
 &= \frac{c \cdot k \cdot k}{\frac{1}{s} \cdot c \cdot k \cdot k + \frac{s-1}{s} \cdot d \cdot d} \approx \frac{s \cdot c}{s+c-1} \approx s
 \end{aligned} \tag{1}$$

The sizes of $d \times d$ and $k \times k$ are similar, and $s \ll c$. In the same way, the compression ratio can be calculated as follows:

$$r_c = \frac{n \cdot c \cdot k \cdot k}{\frac{n}{s} \cdot c \cdot k \cdot k + (s - 1) \cdot \frac{n}{s} \cdot d \cdot d} \approx \frac{s \cdot c}{s + c - 1} \approx s \quad (2)$$

where c is the number of channels of the input feature map, k is the kernel size, the averaged kernel size of each linear operation is $d \times d$, h' is the height of the output feature map, w' is the width of the output feature map, n is the number of channels of the output feature map, and s refers to the number of Ghost features, which is the same as the speedup ratio.

We used the Ghost module to design improved Ghost bottlenecks, as shown in Figure 5.

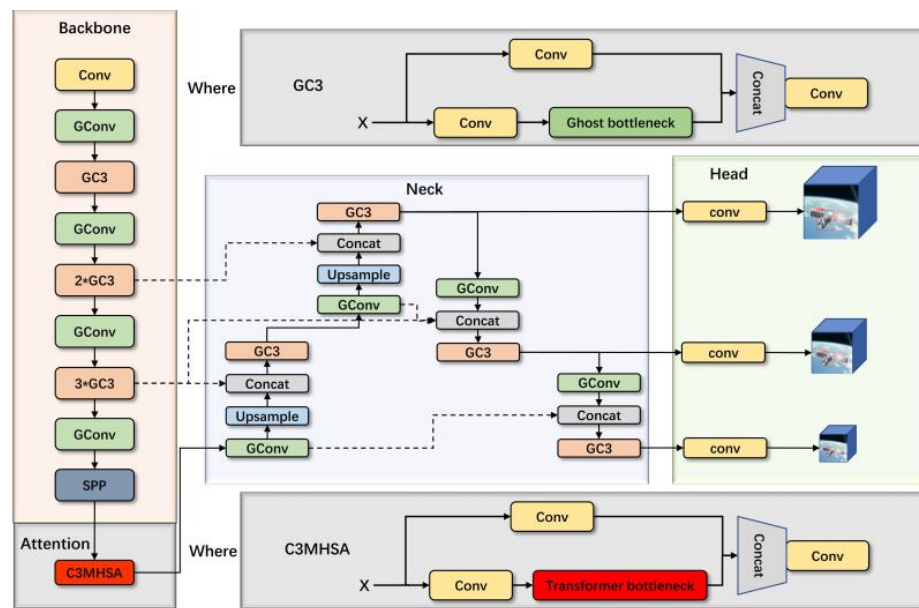


Figure 4. Detection algorithm structural diagram.

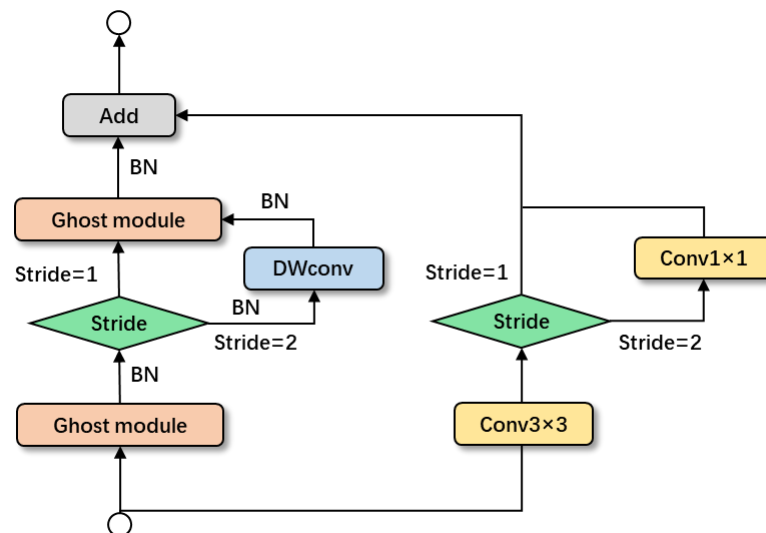


Figure 5. Improved Ghost bottleneck structure. If stride = 1, the two Ghost modules are directly connected, and a 3×3 convolution block is used by shortcut. If Stride = 2, the two Ghost modules are connected by depthwise convolution, and a 3×3 convolution block and a 1×1 convolution block are added in the shortcut.

The initial few Ghost modules in Ghost bottlenecks have minimal dimensions as a result of the channel compression we utilize, which significantly reduces the breadth of the model. The ReLU activation function was removed from the first Ghost module because MobileNetV2 suggests that it will obliterate data in low-dimensional space. We performed experiments and evaluated the outcomes.

The experiment demonstrates that the average accuracy (mAP) is 1.1% greater than that of the original activation function once the ReLU activation function of the first five Ghost modules is removed. The detection speed on the GPU is 1.5 ms faster than that on the CPU, which results in a 6% reduction in detection time. Utilizing a linear layer is essential because it stops nonlinearity from erasing information, according to experimental findings.

We may increase the generality and accuracy of the method, as well as the utilization of features, by adding a convolution layer to the shortcut of the residual block.

3.3. Improvement in Detection Accuracy

The movement of the spacecraft and the rotation of the Earth have an impact on the identification of spacecraft components. Because of interference from changes in the brightness, shape, and Earth backdrop of spacecraft components, detection algorithm performance suffers. This study suggests a combination of CNN and Transformer models, employing CNN to extract features and MHSA to collect global information, enabling detection algorithms to recognize spacecraft components under complicated disturbances. The combination of CNN and Transformer models is inspired by DETR [25] and BoTNet [26]. Figure 6 depicts the MHSA's organizational structure. The MHSA module has the potential to mine feature representations, attain global self-attention, and enhance the capacity to acquire diverse information by using a self-attention mechanism. Due to the low resolution of feature maps at the end of the backbone, to reduce the computation cost, we use MHSA in the last CSP module of the backbone. Since the convolutional layer has the characteristics of weight sharing, we can use the pointwise convolutional layer instead of the linear layer to reduce the parameters of the model.

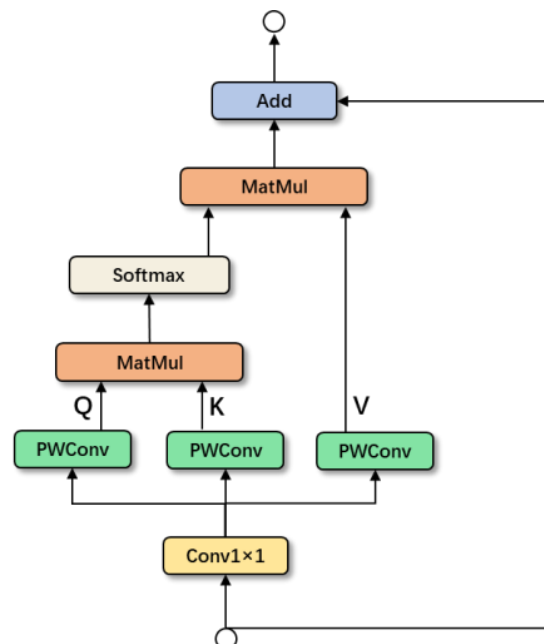


Figure 6. Structural diagram of multi-head self-attention.

In the inspection task of spacecraft components, multiscale inspection is very important, especially for smaller spacecraft components. However, the existing path aggregation network (PANet) does not make good use of the extracted features, and the weights of features with different sizes are the same, which makes it difficult to meet the requirements

of small object detection. However, BiFPN can perform multiscale feature fusion easily and quickly without adding too much cost. It introduces learnable weights to learn the importance of different input features, and repeatedly applies top-down and bottom-up multiscale feature fusion. This structure makes full use of features of different sizes, has strong semantic information, and can improve the accuracy of multiscale object detection. The structures of PANet and BiFPN are shown in Figure 7.

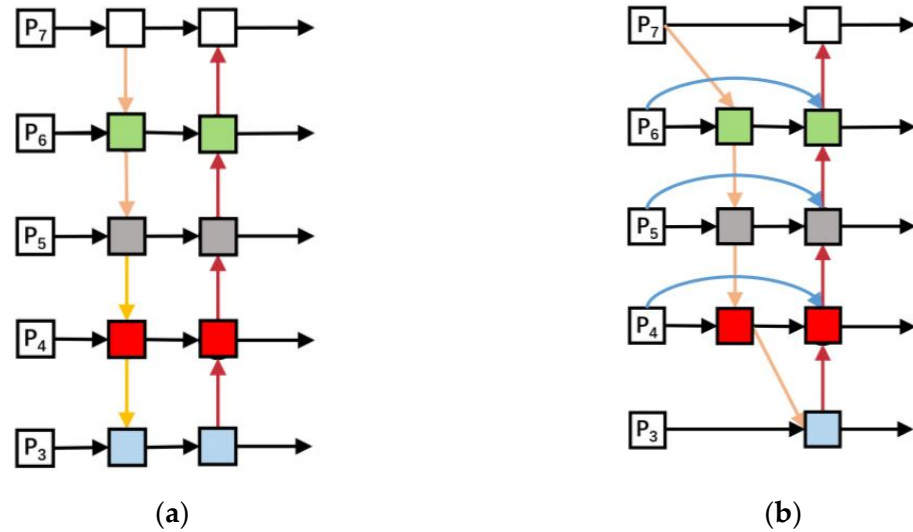


Figure 7. Structural diagram of (a) PANet and (b) BiFPN.

3.4. Loss Function

The loss function includes the following three types of losses: confidence loss function L_{object} , bounding box loss function L_{box} , and classification loss function L_{class} . The loss function is the weighted sum of three losses, as shown in the following equation:

$$L = \lambda_1 L_{object} + \lambda_2 L_{box} + \lambda_3 L_{class} \tag{3}$$

The confidence loss function L_{object} calculates binary cross-entropy for the confidence score in the prediction box and the IOU values of the prediction box and the ground truth box. The classification loss function is similar to the confidence loss, and the classification loss is calculated by the category score of the prediction box and the the ground truth box. The bounding box loss function L_{box} uses CIoU to take into account the distance, overlap rate, scale, and aspect ratio between the target and anchor to make the bounding box more stable, encourage fast convergence, and improve performance. Among them, λ_1 , λ_2 , and λ_3 represent the weights of the three loss functions in the total loss function. The mathematical expression of the CIoU function is shown as follows:

$$L_{CIoU} = 1 - IoU + \frac{\rho^2(b, b_{gt})}{e^2} + \alpha v \tag{4}$$

The variables are shown in the following formulae:

$$IoU = \frac{|B \cap B_{gt}|}{|B \cup B_{gt}|} \tag{5}$$

$$v = \frac{4}{\pi^2} \left(\arctan \frac{w_{gt}}{h_{gt}} - \arctan \frac{w}{h} \right)^2 \tag{6}$$

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{7}$$

where B and B_{gt} represent the predicted box and the ground truth, respectively; e represents the diagonal length of the bounding box; ρ represents the center point distance between the predicted box and the ground truth; b and b_{gt} represent the center point of the predicted box and the ground truth, respectively; h and w represent the height and width of the predicted box, respectively; and h_{gt} and w_{gt} represent the height and width of the ground truth, respectively.

We use YOLOv5 as the framework; the backbone is fused with MHSA, and BiFPN is used as the neck. The algorithm is compressed by incorporating the improved Ghost module and channel compression method, named YOLOv5-GMB. In the experiment, we compare the lightweight algorithms of different backbones to show the advantages of the algorithm in terms of the number of parameters, mAP, and detection speed. At the same time, we also verify the effectiveness of the improved strategy.

4. Experimental Results and Discussion

In this paper, the strategy of transfer learning is used in the experimental process. First, the COCO dataset is used to pretrain our proposed network model, and the weight parameters of the pretrained model are obtained. Then, the weight parameters of the pretrained model are fine-tuned using the constructed spacecraft dataset. Thus, the detection of spacecraft components is completed.

We use the MobileNetv3, ShuffleNet2, and GhostNet algorithms to replace the backbone of YOLOv5 and adjust the neck accordingly. The lightweight detection algorithms, including YOLOv5-MobileNetv3, YOLOv5-ShuffleNet2, and YOLOv5-GhostNet, are obtained to train with the constructed spacecraft components dataset.

4.1. Evaluation Index

The performance of a model is usually evaluated using precision (P), recall (R), mean average precision (mAP), and frames per second (FPS), where $P = TP / (TP + FP)$. The TP and FP all represent the numbers of tested positive samples, but FP is actually negative. Here, $R = TP / (TP + FN)$, where FN represents the number of samples that are positive but tested negative. Moreover, Equation (8) is as follows:

$$mAP = \frac{\sum_{i=1}^C AP_i[t]}{C} \quad (8)$$

where t is the IOU threshold; Equation (9) is as follows:

$$AP[t] = \int_0^1 P dR \quad (9)$$

where C represents the total number of classes.

Here, FPS (frames per second) is the number of images that can be processed per second.

Precision indicates the ratio of the number of correctly predicted positive samples to the number of all predicted positive samples. Recall represents the ratio of the number of correctly predicted positive samples to the total number of true positive samples. Finally, mAP represents the average AP of all categories within all images.

4.2. Experimental Details

The hardware configuration of the experimental environment is as follows: the CPU is an AMD R7 5800 H, the GPU is an NVIDIA RTX 3070, the operating system is Windows 10, and the compilation environment is Python 3.8 + PyTorch 1.8.0 + CUDA 11.3. In the experiment, the batch size (the number of data samples in a batch training) is 8, the image size is 640×640 , the epoch is 2000, and the initial learning rate is 0.001. To achieve high accuracy of the training model, the transfer learning strategy is used to pretrain the model on the COCO dataset and then train the model on the constructed spacecraft components dataset. We use the warm-up strategy, which uses a small learning rate to start training

to ensure the stability of the model, and we increase the learning rate after the model becomes stable to make the model converge quickly. Using the cosine annealing strategy, the learning rate is decayed by the cosine function to obtain the optimal model. The change in the loss function in the training process is shown in Figure 8. It can be seen that the network converges quickly.

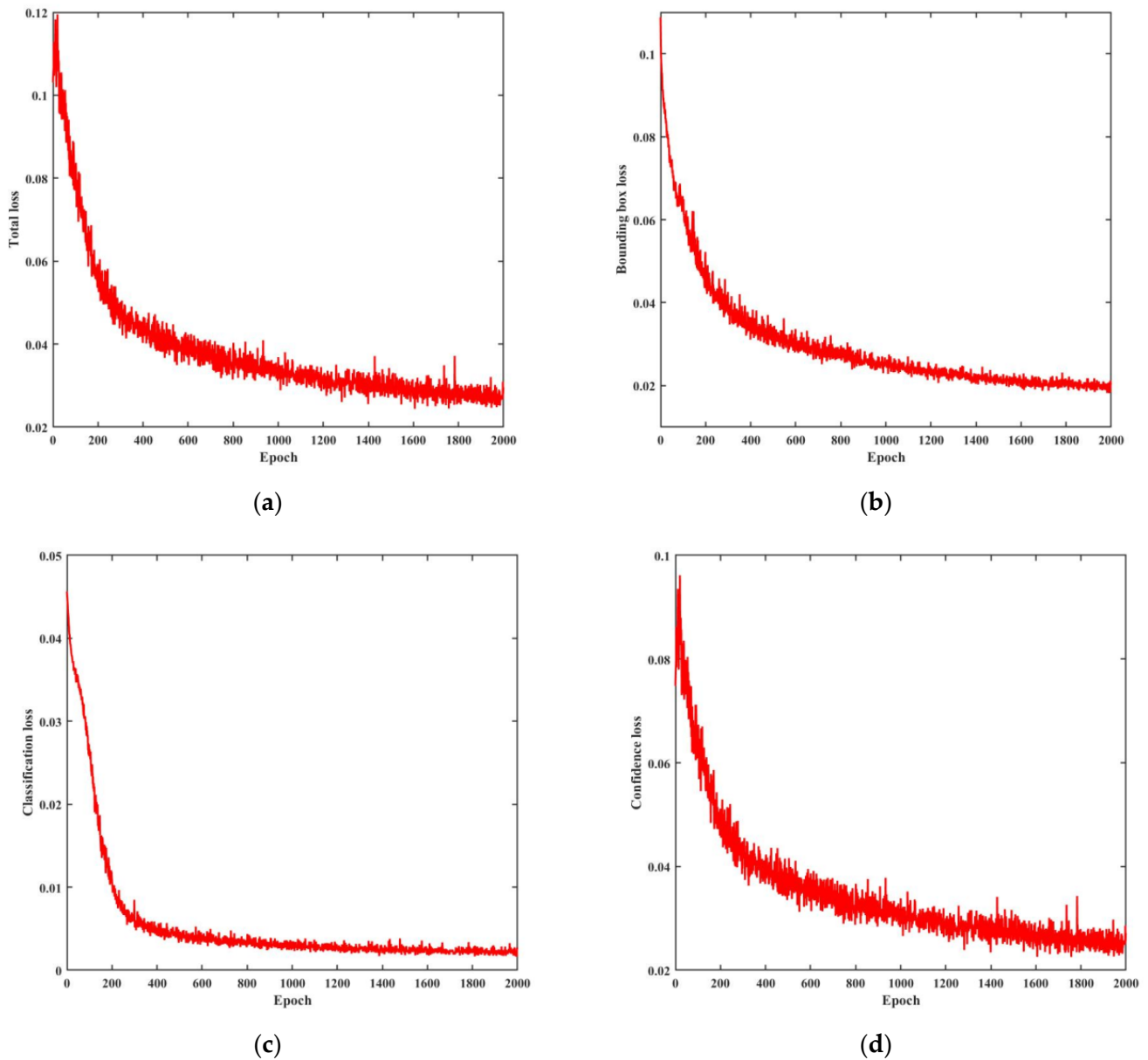


Figure 8. Change in the loss function during training. (a) Total loss. (b) Bounding box loss. (c) Classification loss. (d) Confidence loss.

The algorithm is for the detection of spacecraft components. The main features are the presence of small objects and the large influence of background interference. For the problem of small objects, we regenerate a new anchor box by clustering all labels, which makes the anchor box more suitable for the dataset. At the same time, we use the image scaling method to scale some large images to generate small images, which can enhance the network's detection ability for small objects. For problems with a large impact of background interference, we use data augmentation to make the model predict multiple different versions of the same image for better detection ability.

In this paper, we enhance the luminance, saturation, and noise of the image to accommodate the effects of luminance and noise in space. When dealing with geometric distortions, we added random scaling, cropping, translation, clipping, and rotation. In addition to the above global pixel enhancement method, we also use mosaic data enhancement, using four images for stitching, and each image has its corresponding bounding box. After the four images are stitched together, a new image and the corresponding bounding box of this image are also obtained. The effectiveness of data enhancement lies in expanding the dataset to make the model more robust to images acquired in different environments.

We improve robustness through data augmentation and the model structure. The structure of the algorithm makes full use of multiscale information by introducing BiFPN and generating new anchor boxes suitable for the dataset by aggregating all labels. These methods increase the robustness of the algorithm to scale. We use enhanced data enhancement to perform targeted enhancement according to the imaging characteristics of the spacecraft. This method improves the robustness of the algorithm to disturbance. For the test set, images are acquired in different environments. Therefore, the test results of the algorithm reflect the robustness and generalization performance of the algorithm.

4.3. Test and Results

We use YOLOv5 as the framework to compress the backbone parameters with the fusion of the improved Ghost module and CSPDarknet53, and then train on the COCO dataset to obtain a pretraining model. The evaluation indexes of the model are obtained by training on the spacecraft components dataset. We found that the model has the potential for further compression. We compress the resulting YOLOv5s-Ghost model with 10% steps using the channel compression method. The evaluation indexes of the models are compared. We stop compression at 50% because the model indexes dropped seriously at higher compression ratios. We take the YOLOv5s-Ghost model with a compression ratio of 50% as our lightweight model and name it YOLOv5sl-Ghost.

Due to the obvious decline of indicators, we add MHSA and BiFPN to the YOLOv5sl-Ghost algorithm to improve the detection ability of the algorithm and obtain the YOLOv5-GMB algorithm. Table 1 compares YOLOv5-GMB with other detection algorithms, including YOLOv5s, YOLOv5s Ghost, and YOLOv5sl Ghost.

Table 1. Experimental data indicators of YOLOv5-GMB and other detection algorithms.

Model	Precision	Recall	mAP	Parameter	GFLOPs
YOLOv5s	0.97	0.81	0.86	7.1 M	16
YOLOv5s-Ghost	0.94	0.76	0.81	3.7 M	8
YOLOv5sl-Ghost	0.9	0.68	0.74	0.9 M	2.3
YOLOv5-GMB	0.95	0.71	0.81	1 M	2.4

Through data analysis, we compress the algorithm through the Ghost module and channel compression. Compared with the original YOLOv5s, the number of model parameters is reduced by 87%, the number of model calculations is reduced by 85%, and the total number of parameters is only 0.9 m. The detection time is reduced by 65% and 22% on the CPU and GPU, respectively, which improves the detection speed and greatly reduces the number of model parameters. The parameters of the YOLOv5-GMB model are slightly increased, but the accuracy is improved by 7%, which is roughly comparable to the performance of YOLOv5s-Ghost with 3.6 times larger parameters.

4.4. Ablation Experiments

In this paper, ablation experiments are designed based on YOLOv5sl Ghost and combined with different improvement strategies. The experimental results are shown in Table 2.

Table 2. Ablation study on spacecraft dataset.

Model	Precision	Recall	mAP	Parameter	GFLOPs
YOLOv5sl-Ghost	0.9	0.68	0.74	0.9 M	2.3
YOLOv5sl-Ghost-MHSA	0.94	0.70	0.78	1 M	2.3
YOLOv5sl-Ghost-BiFPN	0.92	0.68	0.77	0.9 M	2.3
YOLOv5sl-Ghost-MHSA-BiFPN	0.95	0.71	0.81	1 M	2.4

To reduce the impact of model size and model parameter decline on the detection ability of spacecraft components, this paper improves the ability to capture different information by introducing MHSA and BiFPN to fuse information of different scales. Through experiments, it is found that introducing the MHSA and BiFPN model to the YOLOv5sl-Ghost model slightly increases the number of parameters and computations. However, the mAP is improved by 7% compared with that when it is not used, which is almost the same performance as YOLOv5s-Ghost. Therefore, the increase in the computation amount is worthwhile.

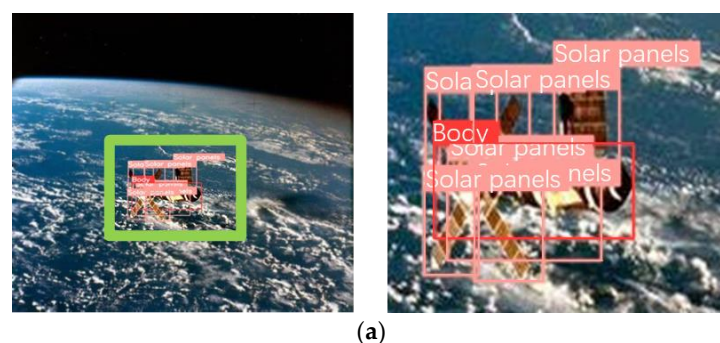
4.5. Experimental Results of the Comparative Method

Existing lightweight detection algorithms, such as MobileNetv3, ShuffleNetv2, and GhostNet, have excellent performance in detection ability and are also the main backbones of lightweight algorithms. We replace the backbone of YOLOv5 with the MobileNetv3, ShuffleNetv2, and GhostNet algorithms and adjust the neck accordingly. Under the same environment, the lightweight detection algorithms YOLOv5-MobileNetv3, YOLOv5-Shufflenetv2, and YOLOv5-GhostNet are obtained by training on the constructed spacecraft dataset and compared with the algorithm in this paper. The evaluation indicators of the models are compared. Table 3 lists the model evaluation indicators and makes a comparison.

Table 3. Comparison of multiple lightweight algorithms on spacecraft datasets.

Model	Precision	Recall	mAP	Parameter	GFLOPs	FPS (CPU)	FPS (GPU)
YOLOv5-MobileNetv3	0.91	0.69	0.77	1.5 M	3.6	13	47
YOLOv5-ShuffleNetv2	0.9	0.66	0.76	1.5 M	3.6	9	43
YOLOv5-GhostNet	0.9	0.68	0.76	1.3 M	3.9	14	50
YOLOv5-GMB (Ours)	0.95	0.71	0.81	1 M	2.4	16	53

The experimental results show that the lightweight model YOLOv5-GMB in this paper is superior to YOLOv5-MobileNetv3, YOLOv5-ShuffleNetv2, and YOLOv5-GhostNet in terms of model parameter amount, model calculation amount, mAP, and FPS. The model parameter amount is only 1 M, and the model calculation amount is only 2.4 GFLOPs. Compared with the lightweight backbone YOLOv5 algorithm, the algorithm has the smallest model, the best detection performance, and the highest FPS. Figure 9 shows the experimental results of the comparison method under fully illuminated conditions. Figure 10 shows the experimental results of the comparison method under partially shadowed conditions. The algorithm proposed in this paper can accurately detect all spacecraft components.

**Figure 9.** Cont.

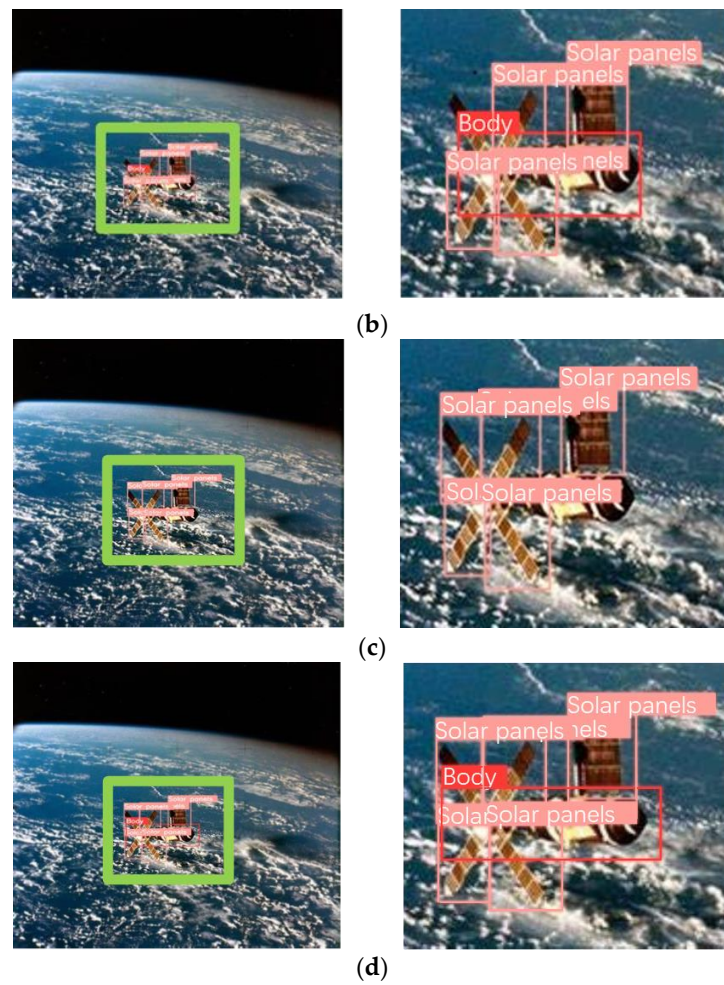


Figure 9. The experimental results of the comparison method under fully illuminated conditions. The left is the detection image. The right is a close-up of the test results. (a) YOLOv5-MobileNetv3 detection results. (b) YOLOv5-ShuffleNetv2 detection results. (c) YOLOv5-GhostNet detection results. (d) YOLOv5-GMB detection results.

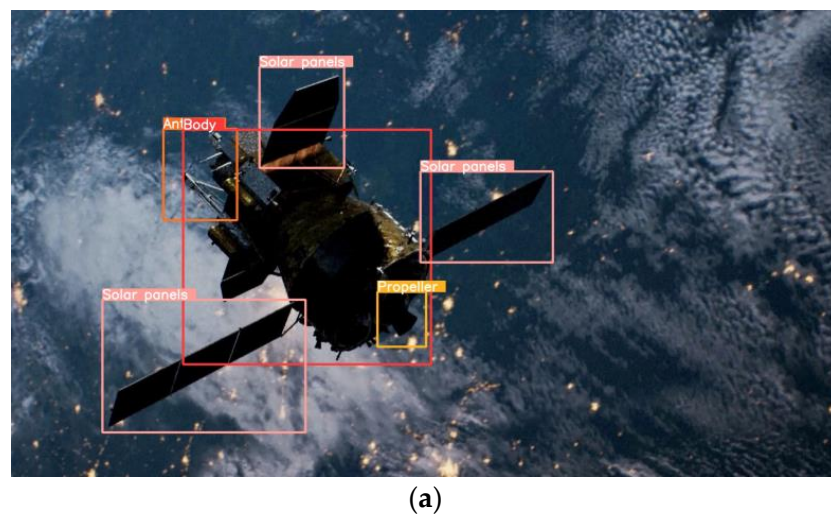


Figure 10. Cont.

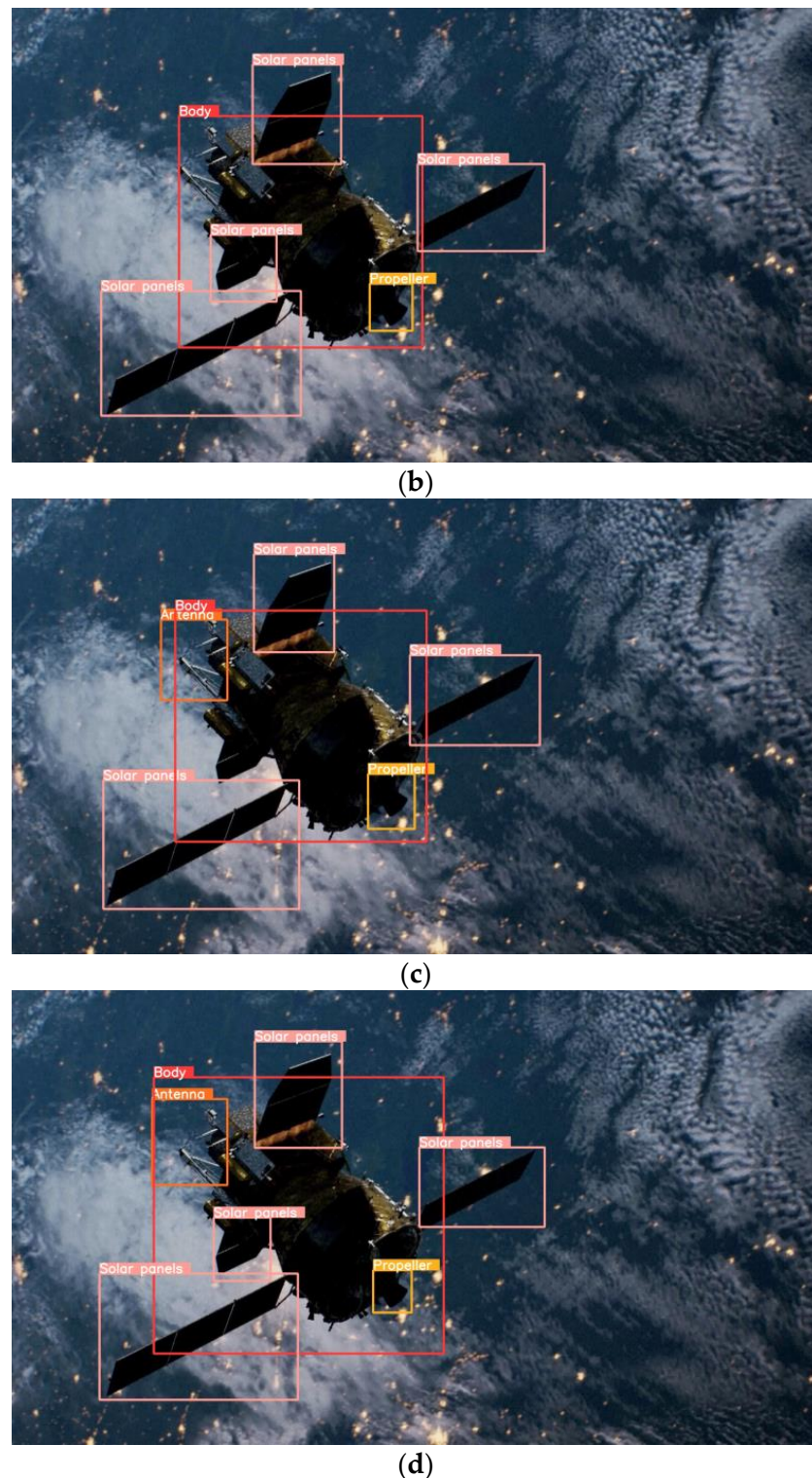


Figure 10. The experimental results of the comparison method under partially shadowed conditions. (a) YOLOv5-MobileNetv3 detection results. (b) YOLOv5-ShuffleNetv2 detection results. (c) YOLOv5-GhostNet detection results. (d) YOLOv5-GMB detection results.

5. Conclusions

In this paper, a lightweight spacecraft component detection method is proposed. We use the Ghost module and channel compression methods for their lightweight quality and improve the algorithm according to the characteristics of spacecraft images. The proposed YOLOv5 GMB only uses 1 M parameters to maintain a high precision of 0.95. The compar-

tive experimental results show that, compared with the traditional lightweight detection model, the proposed YOLOv5-GMB achieves better detection results with fewer parameters and calculation amounts, which can effectively improve the detection efficiency while reducing the model weight. However, under a more complex environment, our detection performance cannot be guaranteed. This is also the direction of our future research.

Author Contributions: Conceptualization, Y.L. and X.Z.; methodology, Y.L., X.Z. and H.H.; validation, formal analysis and investigation: Y.L., X.Z. and H.H.; writing—original draft preparation, Y.L.; writing—review and editing, Y.L. and X.Z.; supervision, X.Z. and H.H.; project administration, X.Z. and H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Basic Scientific Research Project, grant number JCKY2020903B002, and the National Natural Science Foundation of China, (grant number) 51827806.

Data Availability Statement: Not applicable.

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

References

1. Wen, C.X.; Qiao, D. Calculating Collision Probability for Satellite Long-term Encounters Through the Reachable Domain Method. *Astrodynamics* **2022**, *6*, 141–159. [\[CrossRef\]](#)
2. Qi, Y.; Qiao, D. Stability Analysis of Earth Co-orbital Objects. *Astron. J.* **2022**, *163*, 211. [\[CrossRef\]](#)
3. Cui, P.Y.; Qiao, D. State-of-the-art and prospects for orbital dynamics and control near small celestial bodies. *Adv. Mech.* **2013**, *43*, 526–539. (In Chinese)
4. Da Fonseca, I.M.; Goes Luiz, C.S. Attitude dynamics and control of a spacecraft like a robotic manipulator when implementing on-orbit servicing. *Acta Astronaut.* **2017**, *137*, 490–497. [\[CrossRef\]](#)
5. Li, X.Y.; Qiao, D.; Huang, J.C.; Han, H.W.; Meng, L.Z. Dynamics and control of proximity operations for asteroid exploration mission. *SCIENTIA SINICA Phys. Mech. Astron.* **2019**, *49*, 69–80. (In Chinese) [\[CrossRef\]](#)
6. Cui, P.Y.; Qiao, D.; Cui, H.T.; Luan, E.J. Target Selection and Transfer trajectory design for asteroid exploration. *Sci. Sin. Phys. Mech. Astron.* **2010**, *40*, 677–685. (In Chinese)
7. Han, H.W.; Qiao, D.; Chen, H.B. Optimization of Aeroassisted Rendezvous and Interception Trajectories between Non-Coplanar Elliptical Orbits. *Acta Astronaut.* **2019**, *163*, 190–200. [\[CrossRef\]](#)
8. Qiao, D.; Zhou, X.Y.; Zhao, Z.D.; Qin, T. Asteroid approaching orbit optimization considering optical navigation observability. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *99*, 1. [\[CrossRef\]](#)
9. Caruso, A.; Quarta, A.A.; Mengali, G.; Bassetto, M. Optimal On-Orbit Inspection of Satellite Formation. *Remote Sens.* **2022**, *14*, 5192. [\[CrossRef\]](#)
10. Volpe, R.; Cenci, C. Optical-aided, autonomous and optimal space rendezvous with a non-cooperative target. *Acta Astronaut.* **2019**, *157*, 528–540. [\[CrossRef\]](#)
11. Xu, G.L.; Xu, J. Spacecraft target recognition based on illumination fuzzy similarity fusion invariant moment. *J. Aeronaut.* **2014**, *35*, 857–867.
12. Chen, L.; Huang, P.F. Spatial non cooperative target detection based on improved HOG feature. *J. Aeronaut.* **2016**, *37*, 717–726.
13. Zhi, X.Y.; Hou, Q.Y. Optical recognition of typical space-based targets based on multi feature fusion. *J. Harbin Inst. Technol.* **2016**, *48*, 44–50.
14. Wang, L. Research on spatial multi-target recognition method based on deep learning. *Unmanned Syst. Technol.* **2019**, *2*, 49–55.
15. Li, L.Z.; Zhang, T. Feature detection and recognition of spatial non cooperative targets based on deep learning. *J. Intell. Syst.* **2020**, *15*, 1154–1162.
16. Zeng, H.Y.; Xia, Y. Space Target Recognition Method Based on Deep Learning. In Proceedings of the 20th International Conference on Information Fusion (Fusion), Xi'an, China, 10–13 July 2017; pp. 1–5.
17. Chen, Y.; Gao, J.; Zhang, K. R-CNN-based satellite components detection in optical images. *Int. J. Aerosp. Eng.* **2020**, *2020*, 1–10. [\[CrossRef\]](#)
18. Zhu, X.; Lyu, S.; Wang, X. TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, Canada, 10–17 October 2021; pp. 2778–2788.
19. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A. An Image is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4–8 May 2019; pp. 1314–1324.
20. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 10781–10790.
21. Sandler, M.; Howard, A.; Zhu, M. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4510–4520.

22. Howard, A.; Sandler, M.; Chu, G. Searching for mobilenetv3. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–21 June 2019; pp. 1314–1324.
23. Ma, N.; Zhang, X.; Zheng, H.T. Shufflenet v2: Practical guidelines for efficient CNN architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
24. Han, K.; Wang, Y.; Tian, Q. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 1580–1589.
25. Carion, N.; Massa, F.; Synnaeve, G. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Online, 23–28 August 2020; pp. 213–229.
26. Srinivas, A.; Lin, T.Y.; Parmar, N. Bottleneck transformers for visual recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 19–25 June 2021; pp. 16519–16529.