remote sensing

MDPI

Article

SS R-CNN: Self-Supervised Learning Improving Mask R-CNN for Ship Detection in Remote Sensing Images

Ling Jian [1], Zhiqi Pu [1], Lili Zhu [2], Tiancan Yao [1] and Xijun Liang [2,*]

[1] School of Economics and Management, China University of Petroleum, Qingdao 266580, China
[2] College of Science, China University of Petroleum, Qingdao 266580, China
* Correspondence: liangxijunsd@upc.edu.cn

Abstract: Due to the cost of acquiring and labeling remote sensing images, only a limited number of images with the target objects are obtained and labeled in some practical applications, which severely limits the generalization capability of typical deep learning networks. Self-supervised learning can learn the inherent feature representations of unlabeled instances and is a promising technique for marine ship detection. In this work, we design a more-way CutPaste self-supervised task to train a feature representation network using clean marine surface images with no ships, based on which a two-stage object detection model using Mask R-CNN is improved to detect marine ships. Experimental results show that with a limited number of labeled remote sensing images, the designed model achieves better detection performance than supervised baseline methods in terms of mAP. Particularly, the detection accuracy for small-sized marine ships is evidently improved.

Keywords: self-supervised learning; marine ship detection; deep learning; remote sensing images; Mask R-CNN

check for updates

Citation: Jian, L.; Pu, Z.; Zhu, L.; Yao, T.; Liang, X. SS R-CNN: Self-Supervised Learning Improving Mask R-CNN for Ship Detection in Remote Sensing Images. *Remote Sens.* **2022**, *14*, 4383. https://doi.org/10.3390/rs14174383

Academic Editors: Danilo Avola, Alessio Fagioli and Daniele Pannone

Received: 18 July 2022
Accepted: 30 August 2022
Published: 3 September 2022

1. Introduction

Object detection has been widely used in various applications. Typical object detection methods include supervised learning-based object detection methods, self-supervised learning-based methods, and self-attention-based methods. Supervised learning methods have been widely used in remote sensing image detection. U-Net consists of four downsampling and upsampling layers [1], where the introduced skip-connection structure is suitable for distinguishing low-level and high-level semantic information. The sea surface background is the main source of low-level semantic information, while high-level semantic information consists of the deflection, shape of the target ship, wake around the ship, waves and other noise. Another noticeable method is Faster R-CNN, which was proposed by Ross B. Girshick et al. in 2016 [2]. It integrates modules of the feature extraction network, target region localization, and anchor frame regression and classification. Mask R-CNN [3] is a variant of Faster RCNN [2], in which a feature pyramid network is designed to fuse high- and low-level semantic information, and a strategy of ROI alignment is introduced to improve the accuracy of mapping from the anchor frames to the feature maps. Different from the above methods, YOLO [4] is a single-stage object detection model that employs the anchor-free strategy to cut an image into multiple squares. Single-Shot MultiBox Detector (SSD) [5] is another one-stage detection model proposed for the problem of low recall of small objects by YOLO. SSD operates uniform sampling for the aspect ratios and the area ratios on the feature maps and then performs classification and regression of the obtained anchor boxes directly.

Self-supervised-based object detection models mainly utilize contrastive learning strategies. The core idea of contrastive learning is to construct positive and negative sample pairs by data augmentation, and the network learns image representation by maximizing the similarity between positive sample pairs and minimizing the similarity between negative sample pairs. Y. Kalantidis et al. pointed out that the key to improving feature representation in contrastive learning is to construct hard negative sample pairs [6]. Considering that a large negative sample queue is a heavy burden on computational memory, K. He et al. [7] proposed an updatable dictionary to maintain the negative sample queue, and operated contrastive learning by two independent encoders. It outperforms self-supervised models such as BYOL [8] in tested downstream tasks including image classification and image segmentation. Comparatively, SimCLR [9], proposed by T. Chen et al., uses a large batch size to ensure the diversity of negative samples. It also designs a renewable projection head to improve the similarity calculation between the feature vectors.

Self-attention-based methods are another type of object detection model. The concept of a self-attention mechanism originates from the field of natural language processing and is a variant of the attention mechanism that focuses on capturing correlations between internal information while reducing dependence on external information [10]. However, the background of the marine surface is monotonous with simple semantics, and the occurrence of an object does not have much correlation with long-range contextual information. Particularly, more than half of the ships to be detected are small, usually occupying only tens or hundreds of pixels. Hence, this type of method is not quite suitable for ship detection from remote sensing images.

Most of the above works aim to detect the objects of interest in common image datasets, while a few studies have explored strategies of detecting marine ships based on remote sensing images. As marine remote sensing images have rich background noise, targets of low resolutions, heterogeneous angular deflection, and diverse geographic environments, it is typically difficult to learn the intrinsic features of ships. Particularly, in many applications, only a limited number of remote sensing images with target ships are available. Hence, typical object detection schemes cannot be directly applied to marine ship detection from remote sensing images.

Self-supervised learning is an emerging deep learning method; it is a special type of unsupervised learning and relies on prior knowledge existing in nature, such as the connection between zebra stripes and zebras. Self-supervised learning can learn the inherent structure in unlabeled data, such as the logical order of words in a sentence, and the coherence between objects in pictures and the objects themselves. Hence, it is a promising technique for marine ship detection, especially for scenarios where there are only a limited number of labeled remote sensing images. Typical self-supervised learning methods can be grouped into two branches. The first branch is used for dealing with specific applications, such as image restoration and image reconstruction [11,12]. The other branch is designed for learning good feature representation, under which many self-supervised models have been proposed, such as MOCO [7], AMDIM [13], and SimCLR [9]. Their representation networks are all trained by contrastive self-supervised tasks.

In this work, we design a special CutPaste self-supervised approach to train the representation extraction network by normal remote sensing marine images with no ships, based on which a two-stage detection model is designed. The self-supervised learning module learns the feature representations from unlabeled images by cutting and pasting multiple random rectangle areas of the image. The network learns the characteristics of different types of marine ships by identifying the number of cut areas. For the detection module, we employ the Mask R-CNN [3] network structure and make some special improvements. It generates dense and reliable candidate anchor frames through the RPN network and determines accurate detection boxes by the ROI align technique. The proposed method is verified on Airbus datasets. Experimental studies illustrate that the proposed self-supervised object detection method performs better in terms of mAP, AP50, and AP75 than the baseline supervised methods, especially when there is a limited number of labeled images. The main contributions of this work are as follows.

- A self-supervised task is specially designed for small object detection from remote sensing images, which takes advantage of the characteristic that there often exist multiple ships of different sizes located in a common interested image, improving the representative capability of the backbone network.

- A self-supervised-based marine ship detection method is proposed for small-sized remote sensing datasets. Semantic representation learning is accomplished by making full use of unannotated images that contain no detection objects. Hence, the detection method greatly reduces the number of annotated images, improves the detection accuracy, and expands the scope of its application.

2. Method

In marine remote sensing images, many ships to be detected are very small, even only tens of pixels. Hence, it is challenging to learn good representations of these targets. Although the problem could be alleviated by data augmentation [14], the augmented pictures are typically some variation of the original ones, leading to overfitting during network training. Self-supervised-based models attract our attention, as they can make use of unlabeled samples. However, the images lack subdivision information of ship types, making it difficult to design proper comparisons and thus difficult to construct hard negative sample pairs.

In consideration of the characteristics of small target objects in remote sensing images, we propose a marine ship detection model based on an improved CutPaste self-supervised strategy. As shown in Figure 1, the model consists of two modules: the self-supervised learning module and the object detection module. The self-supervised learning module employs a ResNet network trained on the designed CutPaste auxiliary task. The module aims to learn a good representation of unlabeled remote sensing images without detected objects. The object detection module classifies and detects the objects; it is based on a Mask R-CNN network that has been altered so that the FPN network is modified and the mask branch is not used. Our model is named SS R-CNN.

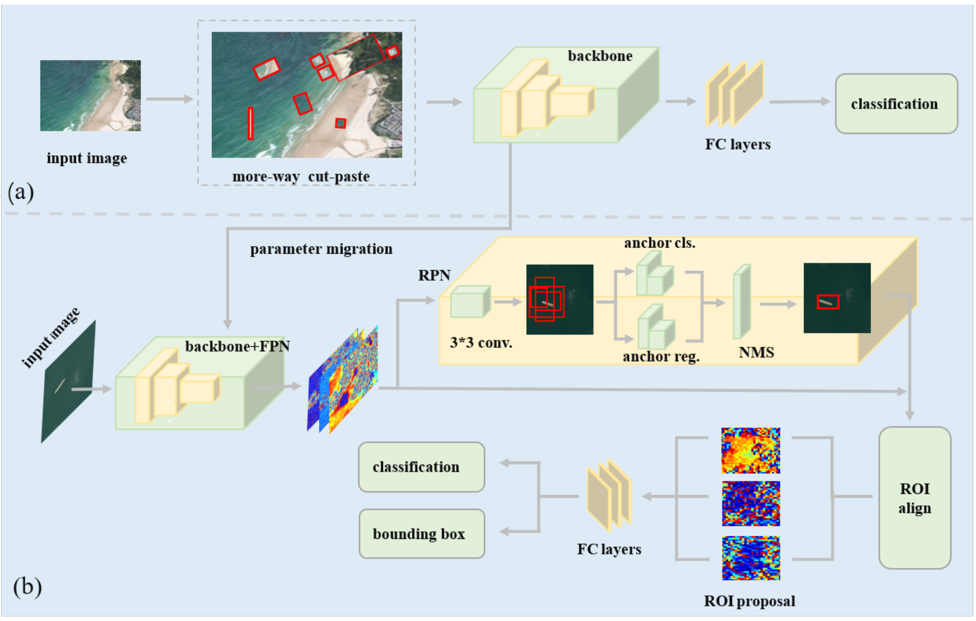


Figure 1. Illustration of the designed self-supervised-based object detection network SS R-CNN. (**a**) Self-supervised learning module. The network is trained by a more-way CutPaste classification task on images with no target objects. (**b**) Object detection module. The representation network migrates from the self-supervised learning module to a modified Mask R-CNN network that generates dense candidate anchor frames through the RPN network and determines the final detection box by the ROI align technique.

2.1. The CutPaste-Based Self-Supervised Learning Module

CutPaste is a self-supervised method proposed for industrial defect detection by Chun-Liang Li et al. [15]. The key point is to construct negative samples by cutting and enhancing some regions of defect-free images and then pasting them, and further training the network by a defect/defect-free binary classification task. Motivated by this work, we design a CutPaste task for remote sensing marine ship detection by multiple cutting operations on a marine image and pasting the rectangles for detection.

As illustrated in Figure 2, there are three types of CutPaste operations in CutPaste data augmentation: block, scar and 3-way. It is not uncommon that multiple ships appear in an object marine remote sensing image. However, detection methods based on representations learned by a typical CutPaste self-supervised learning strategy tend to miss certain ships. Hence, we design a more-way CutPaste self-supervised task in the current work.

(i) The block operation cuts a rectangle area, applies color jittering, and then pastes it onto a random position of the image. In this work, we specially set larger aspect ratios to generate elongated rectangles and set various areas of the rectangles. Specifically, certain small-sized rectangles are generated.

(ii) The scar operation elongates and then rotates the clipping area after a block operation.

(iii) The 3-way operation combines the two above types of operations. It enhances an image randomly by a block or scar operation. The corresponding classifier identifies the image as a normal marine image, an image output by block, or an image generated by scar.

(iv) The more-way operation has the following procedures:
   (a) Select and cut 0–20 rectangles randomly over various areas;
   (b) Perform rotation, color dithering and scaling of the selected rectangle areas;
   (c) Paste the rectangles randomly onto the original image, and the auxiliary learning task aims to detect the number of cut-and-paste rectangles in the image.

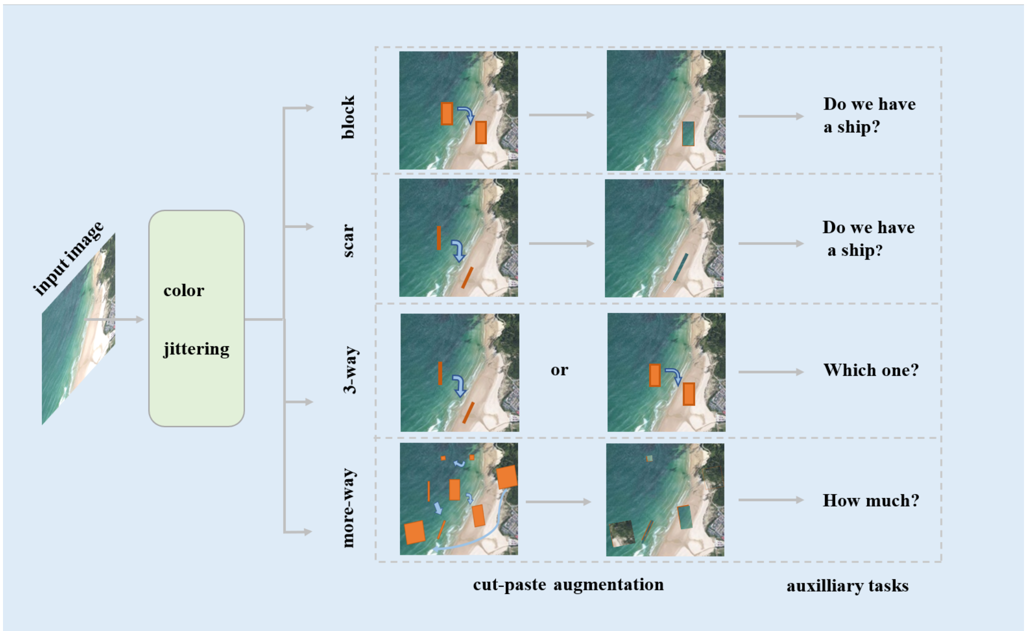


Figure 2. The CutPaste self-supervised tasks. After color jittering, the input image is subjected to various cut and paste operations, and the feature representation network is trained by auxiliary classification tasks. The more-way CutPaste operation is specially designed in this work.

The corresponding learning task of the more-way operation employs the following cross-entropy loss function to measure empirical loss:

$$L_{\text{more-way}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{M} y_{ic}\log p_{ic} \tag{1}$$

where N is the number of training samples, M is the maximum number of cut-and-paste rectangles, p_{ic} is the probability that the ith image has c rectangles, and $y_{ic} = 1$ if the ith sample indeed has c rectangles, otherwise it is 0. The empirical loss defined in (1) employs cross-entropy loss, which considers the number of objects to be predicted. Other loss functions, such as the anomaly score, only indicate whether the image contains anomalies, i.e., whether it contains marine ships of interest in a tested image, meaning small-sized ships may be missed.

In order to ensure that the pretrained network obtained by CutPaste is adaptive to the downstream detection task, we specify standard ResNet-50 as the representation extraction network for the self-supervised learning module. The affine transformation is followed by a Softmax prediction layer served as a classifier to distinguish defects. Moreover, the FC layer is a 10-layer Perceptron, which maps a 512-dimension input vector to a 20-dimension output, as we set the maximum number of ships contained in an image as 20. Then, the number of objects identified is obtained in the form of a probability distribution.

2.2. The Object Detection Module

Considering the requirement of detection accuracy, a two-stage object detection model, Mask R-CNN [3], is selected and modified as the object detection module; it consists of four main functional components: FPN, RPN, ROI align, and the prediction head, as shown in Figure 1b.

Deeper networks (e.g., ResNet) have deep feature maps that retain the representations of large objects, and shallow feature maps that retain the features of small objects. In order to capture the representation of objects with different scales, mask-scoring R-CNN uses the feature pyramid network (FPN) [16] to fuse the feature maps of different stages of the ResNet network. Through upsampling and nonlinear connection operations, the FPN network outputs five different scales of feature fusion maps, with each feature map layer extracting the main information from the previous layer; the fused multi-layered feature maps are beneficial for detecting targets of different scales.

In this work, many target ships are small in the remote sensing images (more than half of the objects are less than 1000 pixels in the tested datasets). As small target objects are mainly represented by high-resolution feature maps, we preserve only the P2–P4 FPN layers, while the P5–P6 layers are removed to simplify the network structure, as illustrated in Figure 3.

The prediction head is another sub-network of Mask R-CNN; it consists of three main branches: category prediction, bounding-box regression and segmentation prediction. It is observed that the background of the sea surface is typically large, and the area of the target objects is usually small, even as small as ten pixel points. Hence, accurate segmentation annotation is usually unobtainable. Therefore, we do not employ the mask prediction task and use only the classification task and bounding-box regression tasks. The empirical loss of the detection model is simplified as

$$L = L_{cls} + L_{box}$$

where L_{cls} and L_{box} are the empirical loss induced by the classification task and bounding-box regression tasks, respectively, and the cross-entropy loss functions are used for both of the tasks. Moreover, to avoid invariance of hyperparameters due to the categories (with/without ship) of the trained instances used by CutPaste self-supervised training tasks and the modified Mask R-CNN, we only migrated the pretrained ResNet50 backbone network parameters and trained the FPN network, RPN network, and ROI head end-to-end with samples containing one or multiple ships.

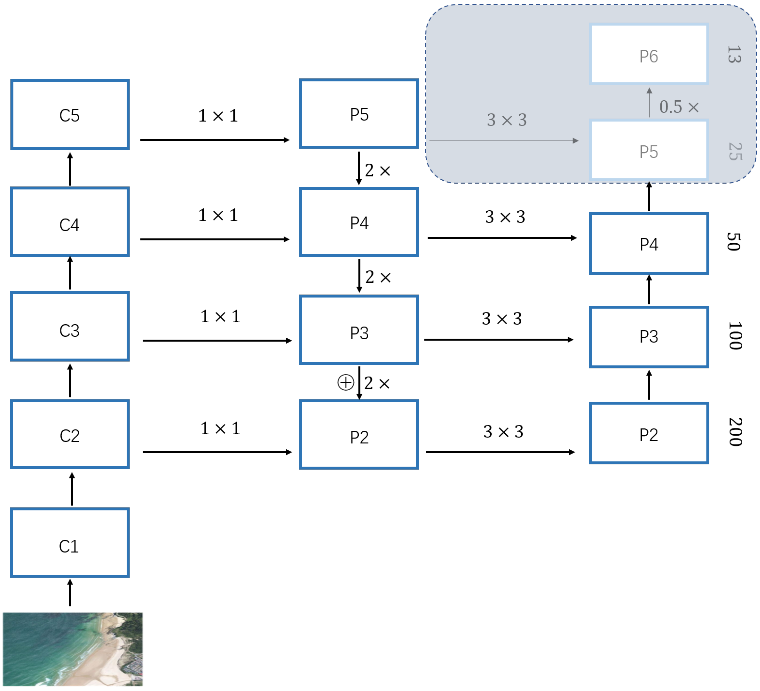


Figure 3. The framework of FPN. C1-C5 indicate ResNet convolution layers; 1 × 1 and 3 × 3 denote 1-dimensional convolution and 3-dimensional convolution operations, respectively; 2× denotes 2-fold upsampling, and 0.5× denotes 0.5-fold downsampling, i.e., max-pooling; ⊕ stands for the element-wise summing operation; P2–P6 denote the output fused feature maps; 200, 100, 50, 25, and 13 are the corresponding dimensions of feature maps of P2–P5.

3. Results

In this section, we evaluate the performance of the proposed object detection method, especially in scenarios where there are only a limited number of labeled images. The base self-supervised-learning experiments are implemented in Python 3.7 on a Linux sever with one Tesla V2 GPU. Then, we finetune our model on colab with one Tesla P100 GPU. The code and pretrained weights can be found on https://github.com/REAL-Madrid01/Remote_GDJC, accessed on 13 August 2022.

3.1. *Dataset and Platforms*

For evaluating the designed method, the Airbus ship detection dataset [17] is selected; it consist of 150,000 remote sensing marine images containing no ships and about 81,724 images containing one or multiple ships and the corresponding detection frames. The images have various backgrounds, such as marine surfaces, offshore, harbors, fog, and waves. The raw dataset is of RLE (run-length decoding) format. To make it suitable for the COCO evaluation protocol [18], the images were transformed into the COCO objection segmentation format using the tool FiftyOne, with the horizontal bounding box used as the format of the detection frames. Figure 4 illustrates the images with captured ships of various sizes. About 10% of the ships in the dataset occupy large-sized areas, while small-sized ships account for more than half of the images, and the remaining 30% of ships are middle-sized.

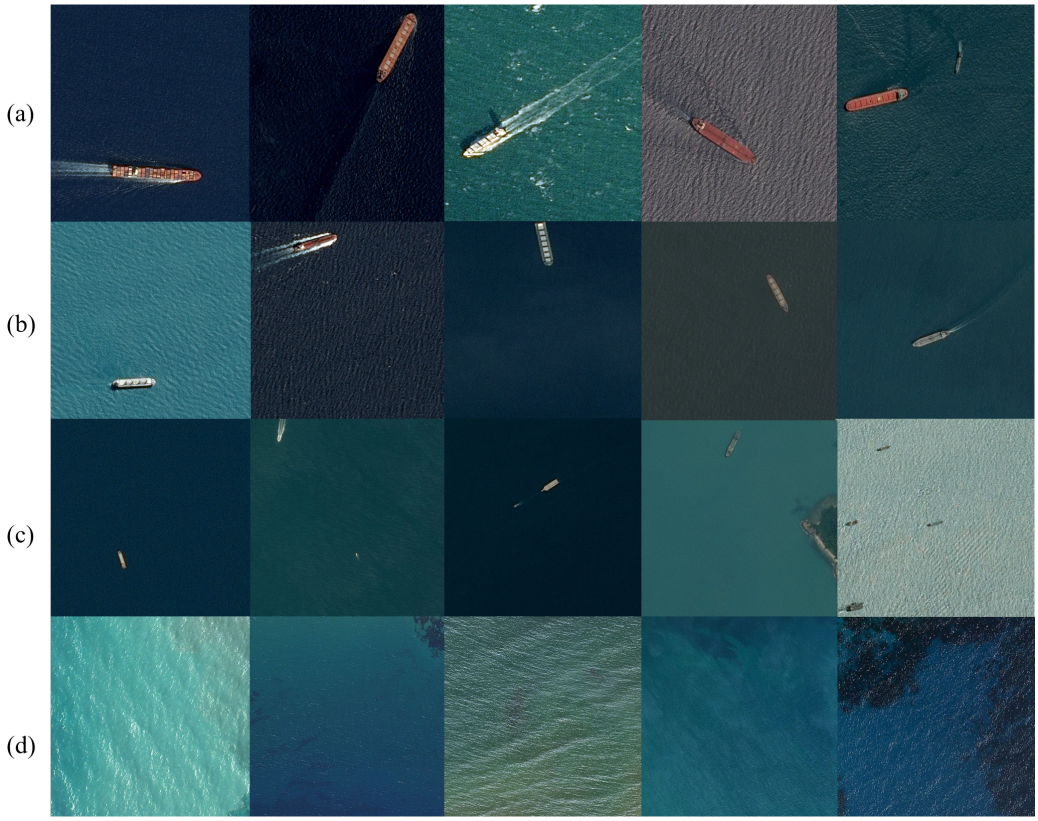


Figure 4. Illustration of the Airbus dataset. The captured ships take up various sizes in the image area, with (**a**) large ships occupying areas more than 96 × 96 pixels, (**b**) medium-sized ships occupy areas between that of the large and small ships, (**c**) and small ships occupy areas less than 32 × 32 pixels; (**d**) images of marine surface with no ships.

3.2. Experimental Setup

For training the self-supervised learning module, 7500 images with no vessels are randomly selected. To evaluate the object detection model, especially for the scenarios of limited labeled images, a series of a specified number of images is randomly sampled as the training set. Meanwhile, the SGD optimizer is used, and the training batch size is set as 8. The learning rate is set to be 0.0005. In the following RPN network, we select three anchor box sizes: 128 × 128, 256 × 256, and 512 × 512, which correspond to the P2–P4 feature maps generated by the FPN network, respectively.

The anchor box step size is set to 4, 8, and 16, respectively. The anchor box aspect ratio is randomly selected from 0.5, 1.0, 2.0. An anchor box is considered a foreground region of interest if the calculated IoU metric of the anchor box and ground-truth box is greater than 0.6. If the IoU metric is less than 0.3, the anchor box is considered the irrelevant background region. After the NMS operation, up to 2000 candidate anchor boxes at most are selected per image.

We employ commonly used metrics in the COCO evaluation protocol to evaluate the proposed model: mAP, AP50, AP75, APs, APm, and APl. The characteristics of the metrics are listed in Table 1; mAP is a primary metric and AP indicates the AP value for small objects. For each of the listed metrics, a larger metric value indicates better detection performance.

Table 1. The employed evaluation metrics.

Metric	Characteristic
mAP	the average AP values, which are calculated with the IoU thresholds located in the interval $[0.5, 0.95]$ with step size 0.05
AP50	the AP value calculated with the IoU metric has a threshold of 0.50
AP75	the AP value calculated with the IoU metric has a threshold of 0.75
APs	the AP value for small objects (occupied area $< 32^2$)
APm	the AP value for medium objects (occupied area is located in the interval $[32^2, 96^2]$)
APl	the AP value for large objects (occupied area $> 96^2$)

3.3. *Comparison with the Baseline Methods*

We first compare the proposed method, SS R-CNN, with two supervised models, SSD [5], a one-stage object detection model, and Mask R-CNN, a two-stage model. To illustrate the role of the designed self-supervised learning module, we fix the detection module and substitute the more-way CutPaste module with other self-supervised learning modules, including MoCo and SimCLR. For training the self-supervised learning module, 7500 images with no vessels are randomly selected; 1000 labeled images, which contain ships, are randomly selected as the training set; and 300 images are randomly selected as the test set. After the training procedure of the model reaches stability, the corresponding detection accuracies are recorded, as listed in Table 2.

Table 2. Comparison with the baseline methods.

	mAP	AP50	AP75	APs	APm	APl
SS R-CNN	0.622	0.758	0.658	0.620	0.723	0.158
Mask R-CNN	0.528	0.688	0.559	0.505	0.649	0.199
SSD	0.257	0.513	0.230	0.127	0.536	0.541
MOCO + Mask R-CNN	0.520	0.698	0.548	0.557	0.589	0.105
SimCLR + Mask R-CNN	0.484	0.657	0.502	0.540	0.550	0.108

(i) It can be seen from Table 2 that the detection accuracies of SS R-CNN are better than those of the other tested supervised methods, Mask R-CNN and SSD, in terms of mAP, AP50, AP75, APs, and APm. As one main difference between SS R-CNN and Mask R-CNN is the designed self-supervised learning module, the results indicate that the self-supervised learning module of SS R-CNN has extracted helpful semantic feature information from the unlabeled images.

(ii) By comparing the accuracies of SS R-CNN with Mask R-CNN pretrained by MOCO and SimCLR (the last two rows of Table 2), it can be seen that the designed more-way CutPaste module more effectively captures feature presentations for the downstream ship detection task.

(iii) For large target objects, the accuracy of SS R-CNN and Mask R-CNN have a large gap compared with SSD in terms of APl. One main reason is that SS R-CNN also employs the Mask R-CNN module, whose detection capacity for large objects is restricted by the number of labeled images. Another known active factor is the size of candidate anchor frames. We discuss these two factors in detail in this subsection.

(iv) The detection performance of small target objects is the main drawback of SSD.

The discovered ships of the tested methods from some typical images are depicted in Figure 5. It can be seen that:

(i) For the images with multiple ships (the first two rows), SS R-CNN correctly detects multiple objects, alleviating the issue of missing vessels;

(ii) SS R-CNN has better performance for small ships and middle-sized ships compared with the supervised methods, SSD and Mask R-CNN, as the predicted frames are more accurate;

(iii) The detected frames of SS R-CNN and Mask R-CNN are not as accurate as that of the SSD method for large ship detection. In the last row of the Figure 5, there is a large ship with a small boat beside it. SSD correctly detects the two objects, while the predicted frames of SS R-CNN and Mask R-CNN are not accurate.

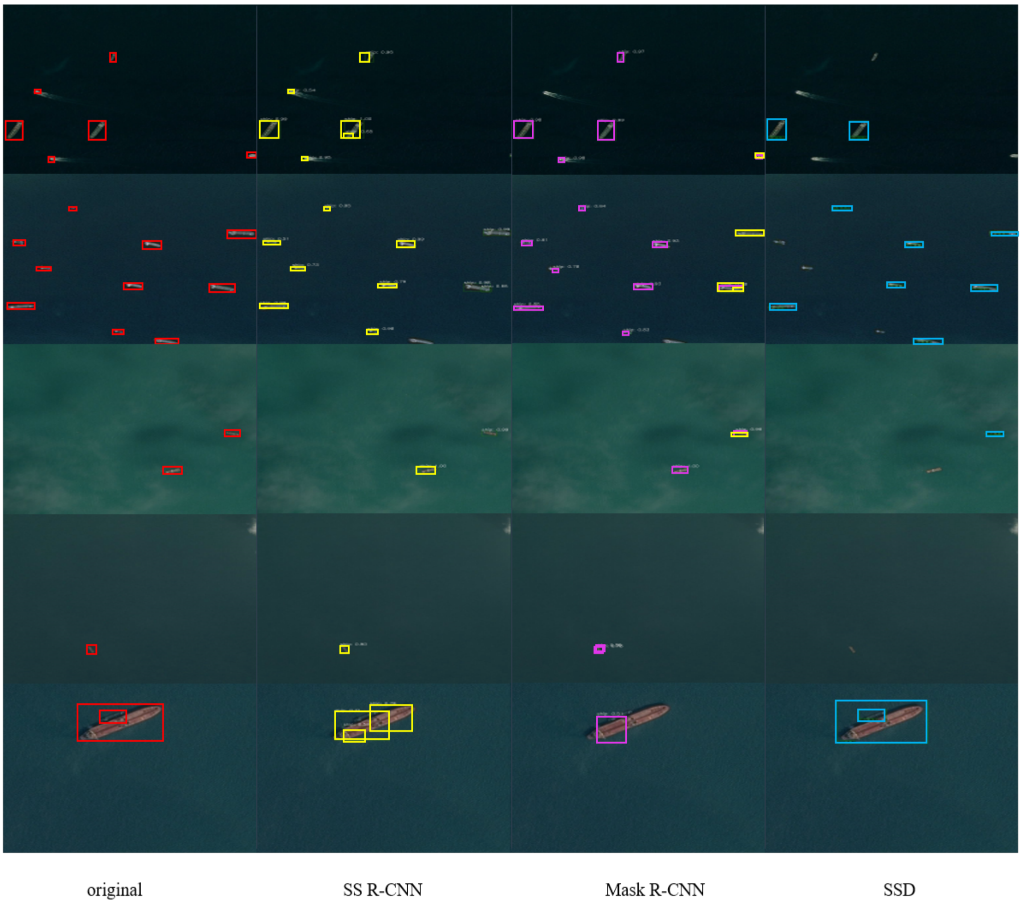


Figure 5. Detected ships of the tested methods. The leftmost column displays several typical remote sensing images, including scenarios with multiple ships (the first two rows), middle-sized ships (the third row), small ships (the fourth row), and large ships accompanied by small boats (fifth row). The following columns display the ships detected by SS R-CNN, SSD, and Mask RCNN, respectively, in different images. The detected ships are marked with rectangles, while the red rectangles in the first column are the ground-truth boxes of the dataset.

Exploring the Reason for Relatively Low Detection Accuracies for Large Ships

In the following, we explore why SS R-CNN and Mask R-CNN have relatively low detection accuracies for large ships. The first possible reason is the lack of training instances with large ships, due to the fact that the proportion of large ships is only 10%, which is quite lower than that of the other types of ships. Another possible reason is that some active hyper-parameters of the object detection module have not been tuned elaborately.

(i) We test SS R-CNN and Mask R-CNN with ratios of small ships: medium ships: large ships of 3:3:4 and 1:1:8. The accuracies are listed in Table 3. Compared with the original case with a ratio of 6:3:1, the detection accuracy for large ships increases about 0.1 from 0.158 in terms of APl when the ratio is set to 3:3:4. Hence, the limited training samples are one reason for the relatively low detection accuracies for large ships. However, as the ratio becomes 1:1:8, the proportion of large ships has increased, but the detection accuracy for large ships no longer increases.

Table 3. Accuracies of SS R-CNN and Mask R-CNN with various ratios of medium ships and large ships.

	mAP	AP50	AP75	APs	APm	APl
SS R-CNN (6:3:1)	0.622	0.758	0.658	0.620	0.723	0.158
SS R-CNN (1:1:8)	0.374	0.573	0.382	0.395	0.453	0.246
SS R-CNN (3:3:4)	0.492	0.671	0.524	0.491	0.617	0.251
Mask R-CNN (6:3:1)	0.528	0.688	0.559	0.505	0.649	0.199
Mask R-CNN (1:1:8)	0.366	0.543	0.384	0.370	0.435	0.260
Mask R-CNN (3:3:4)	0.481	0.664	0.507	0.482	0.596	0.269

(ii) We adjust the sizes of the candidate anchor frames from (32, 64, 128, 256, and 512) to (128, 256, 512, 1024, and 2048); the APl of SS R-CNN improves from 0.158 to 0.267, as shown in Table 4. This indicates that the size of the candidate anchor frames is an active parameter for large ship detection.

Table 4. Accuracies of SS R-CNN with different sizes of the candidate anchor frames.

	mAP	AP50	AP75	APs	APm	APl
SS R-CNN	0.622	0.758	0.658	0.620	0.723	0.158
SS R-CNN (adjusted)	0.588	0.754	0.618	0.621	0.657	0.267

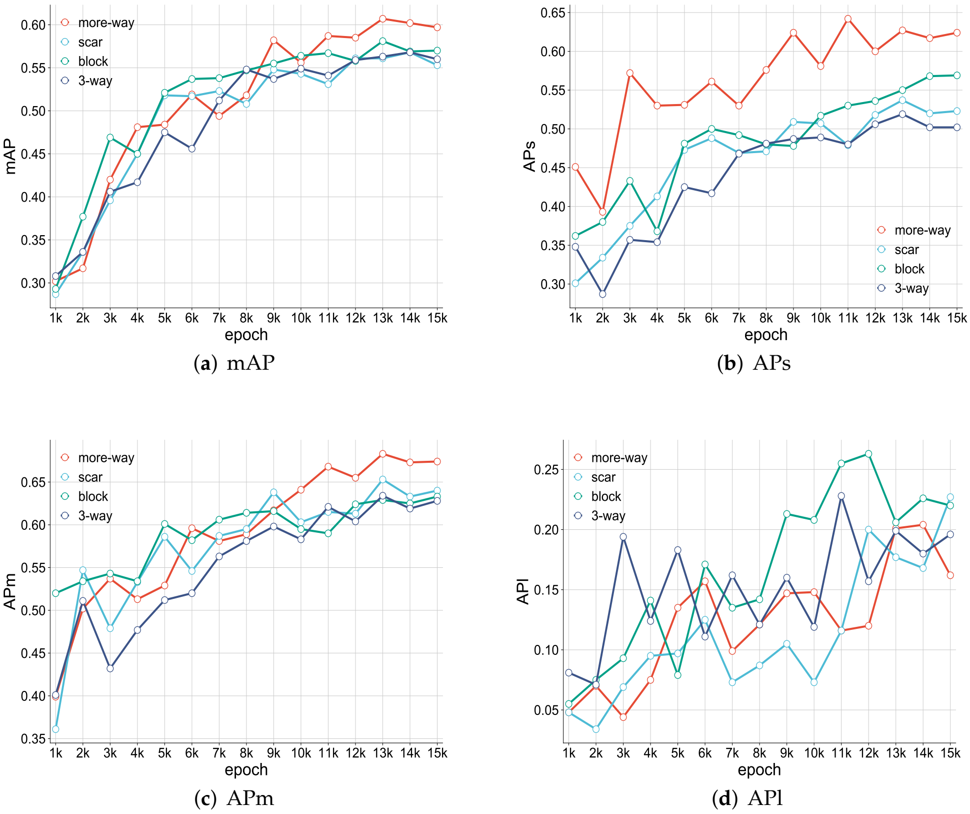
3.4. Employing Different CutPaste Tasks in the Self-Supervised Learning Module

In the self-supervised learning module, a more-way CutPaste task has been designed that allows multiple cut and paste augment operations on a single image. To test the role of the more-way task, we compare it with the typical CutPaste tasks, i.e., block, scar, and 3-way. We randomly select 7500 unlabeled clean marine surface images for training the self-supervised learning network. The accuracies of the downstream object detection module are presented in Table 5, where the numbers in bold indicate the best in terms of the corresponding metrics.

Table 5. Detection accuracy with different CutPaste tasks in SS R-CNN.

	mAP	AP50	AP75	APs	APm	APl
block	0.566	0.733	0.602	0.569	0.694	0.127
scar	0.553	0.697	0.588	0.521	0.698	0.227
3-way	0.560	0.717	0.584	0.537	0.686	0.196
more-way	0.622	0.758	0.658	0.620	0.723	0.158

Table 5 shows that SS R-CNN with the more-way task has better detection accuracy compared with the other self-supervised learning tasks. On the mAP metric, the performance increases about 10.7%, increasing from 0.56 to 0.62. The improvement mainly originates from the amelioration of the accuracy of small ship detection. For the detection of large ships, however, all four tested self-supervised tasks have unsatisfactory detection results, with the APl metric less than 0.30. As the training epochs proceed, the variation of the detection accuracies of SS R-CNN are depicted in Figure 6. The accuracies of mAP, APs, APm, and APl are shown with different CutPaste tasks. In terms of mAP (Figure 6a), APs (Figure 6b), and APm (Figure 6c), SS R-CNN with the more-way CutPaste task is generally superior to that with block, scar, and 3-way pretraining tasks. Particularly, its advantage is more obvious for the detection of small objects (Figure 6b). However, for the detection of large targets (Figure 6d), the accuracies for all four tasks fluctuate below 0.30.



(**a**) mAP (**b**) APs

(**c**) APm (**d**) APl

Figure 6. Variation of the accuracies of SS R-CNN with different CutPaste tasks.

We wonder how the more-way CutPaste task improves detection performance for small ships. Hence, we examine the detected objects of SS R-CNN with various CutPaste tasks. Some typical images with multiple small ships with their detection boxes are selected and are displayed in Figure 7.

Figure 7 shows that SS R-CNN with the more-way CutPaste task catches the target ships more accurately, with fewer missed ships and fewer incorrectly detected objects. Comparatively, SS R-CNN with block, scar, and 3-way preliminary tasks miss more small or tiny objects, which hints to us that the designed more-way CutPaste task is helpful for detecting multiple small objects in remote sensing images.

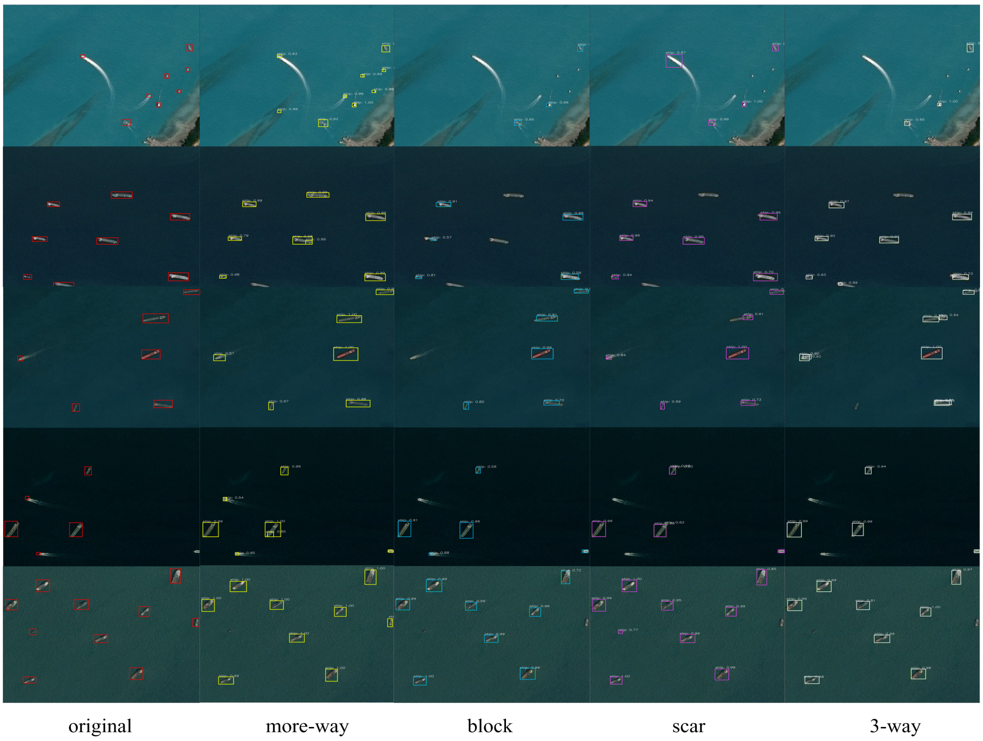


Figure 7. Detected objects by SS R-CNN with various CutPaste tasks in different images. Detection frames of different methods are depicted with different colors.

3.5. Effect of the Number of Labeled Training Images

SS R-CNN equipped with a self-supervised module is designed for the case of limited labeled training images. We are naturally interested in inspecting its performance with varying numbers of labeled training images. Particularly, we select 200–5000 labeled training images, each of which contains one or multiple target objects and corresponding marking boxes. The detection accuracies of SS R-CNN, SSD, and Mask R-CNN pretrained by MOCO are listed in Table 6.

Table 6. Accuracies of SS R-CNN, SSD, and MOCO with varying numbers of labeled training images.

Method	Training Size	mAP	AP50	AP75	APs	APm	APl
SS R-CNN	200	0.476	0.650	0.494	0.515	0.536	0.146
	400	0.514	0.691	0.543	0.546	0.606	0.215
	1000	0.622	0.758	0.658	0.620	0.723	0.158
	2000	0.594	0.754	0.630	0.642	0.657	0.158
	5000	0.598	0.748	0.627	0.661	0.656	0.158
SSD	200	0.179	0.407	0.142	0.105	0.364	0.215
	400	0.222	0.438	0.201	0.121	0.478	0.245
	1000	0.257	0.513	0.230	0.127	0.536	0.541
	2000	0.249	0.492	0.234	0.129	0.512	0.396
	5000	0.297	0.561	0.288	0.156	0.595	0.524
MOCO + Mask R-CNN	200	0.466	0.652	0.476	0.506	0.531	0.112
	400	0.507	0.681	0.537	0.550	0.580	0.107
	1000	0.520	0.698	0.548	0.557	0.589	0.105
	2000	0.557	0.720	0.583	0.604	0.595	0.106
	5000	0.554	0.736	0.584	0.608	0.619	0.116

Table 6 shows that (i) the detection accuracy of each tested method increases overall with the training size, and (ii) SS R-CNN has superior detection accuracy compared with

SSD and Mask R-CNN (pretrained by MOCO) for each training size, except that the detection accuracies for large ships are lower than those of SSD.

3.6. Effects of the Number of Rectangles in More-Way CutPaste Operation for SS R-CNN

The number of maximum CutPaste rectangles is set to five for the more-way CutPaste operation on the Airbus dataset, as marine images with the number of ships located in $[1,5]$ account for 92.2% of the total images with ships. We test SS R-CNN with this parameter set to 10 and 20. The detection accuracies are listed in Table 7, where "#rectangles" indicates the number of rectangles. The results indicate that this parameter can refer to the number of objects occurring in the detected images.

Table 7. Effect of the number of rectangles in more-way CutPaste operation for SS R-CNN.

	mAP	**AP50**	**AP75**	**APs**	**APm**	**APl**
SS R-CNN (#rectangles = 5)	0.622	0.758	0.658	0.620	0.723	0.158
SS R-CNN (#rectangles = 10)	0.519	0.682	0.555	0.537	0.591	0.153
SS R-CNN (#rectangles = 20)	0.543	0.714	0.560	0.594	0.586	0.098

4. Discussion

4.1. Effects of the Input Image Resolution

When the network depth and the size of the convolution kernels are fixed, input images with different resolutions produce feature maps with different resolutions. High-resolution feature maps may improve the detection accuracy for small target objects. We test the performance of SS R-CNN with input image sizes of 1024 × 1024 and 768 × 768. Unfortunately, we do not observe the expected detection improvement. A possible reason is that the images of 1024 × 1024 resolution are upsampled from the 768 × 768 images, which does not essentially improve the legibility.

4.2. Deficiencies of SS R-CNN

The proposed method, SS R-CNN, still has some deficiencies. It has relatively low detection accuracies for large ships with limited labeled training images. According to the preliminary experimental studies, there are at least two influencing factors. The first is the lack of training instances with large ships, considering that the proportion of training images with large ships is only 10% in the experiments. The other is the setting of the hyper-parameter of the sizes of the candidate anchor frames. This limitation of SS R-CNN should be noticed in practical applications.

5. Conclusions

A self-supervised-based object detection model, SS R-CNN, is proposed in this work to detect and localize ships in marine remote sensing images. Specifically, a self-supervised task, i.e., the more-way CutPaste task, is designed for small-object detection from remote sensing images. The main contributions of this research are as follows: (i) A self-supervised module has been specially designed for marine ship detection from remote sensing images. The module efficiently extracted semantic features from unlabeled clear images of marine surfaces, improving the representative capability of the backbone network. (ii) The designed self-supervised-based detection framework greatly reduces the requirement for the number of labeled images, and especially improves the detection accuracy for small ships. Extensive experiments have been conducted, which show that:

(i) The proposed self-supervised learning module can extract semantic features from unlabeled clean images of marine surfaces. Compared to the baseline supervised learning methods, SS R-CNN evidently improved the detection accuracy in the case of a limited number of labeled images. Compared with the best of the baseline supervised methods, SS R-CNN showed 17.8% improvement in terms of mAP and 22.8% improvement in detection accuracy for small target objects.

(ii) Compared with the typical CutPaste tasks, the proposed self-supervised learning module incorporated with the designed more-way CutPaste task can further reduce the number of undetected objects and incorrectly detected objects and, hence, improve the accuracy.

(iii) The proposed self-supervised-based detection framework greatly reduces the requirement for the number of labeled images. For instance, with 200 labeled images, the SS R-CNN achieved an mAP of 0.476.

The preliminary results hint that self-supervised learning methods deserve more attention in scenarios where the collected data samples are insufficient. Moreover, the framework of SS R-CNN is flexible enough to fuse various self-supervised learning models for representation learning, and it is convenient to employ other candidate object detection models for special tasks.

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

Funding: This research was funded by the National Key R&D Program of China (nos. 2021YFA1000100, 2021YFA1000102), and the National Natural Science Foundation of China (no. 61873279).

Data Availability Statement: The proposed model and pretrained weights are available at 13 August 2022. https://github.com/REAL-Madrid01/Remote_GDJC.

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

References

1. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Munich, Germany, 5–9 October 2015; pp. 234–241.
2. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
3. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2961–2969.
4. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
5. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
6. Kalantidis, Y.; Sariyildiz, M.B.; Pion, N.; Weinzaepfel, P.; Larlus, D. Hard negative mixing for contrastive learning. In Proceedings of the Annual Conference on Neural Information Processing Systems, Virtual, 6–11 December 2020; pp. 21798–21809.
7. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 9726–9735.
8. Grill, J.B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Pires, B.A.; Guo, Z.; Azar, M.G.; et al. Bootstrap your own latent—A new approach to self-supervised learning. In Proceedings of the Annual Conference on Neural Information Processing Systems, Virtual, 6–11 December 2020; pp. 21271–21284.
9. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In Proceedings of the 37th International Conference on Machine Learning (ICML), Virtual, 13–18 July 2020; pp. 1597–1607.
10. 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 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
11. Pathak, D.; Krahenbuhl, P.; Donahue, J. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2536–2544.
12. Larsson, G.; Maire, M.; Shakhnarovich, G. Colorization as a proxy task for visual understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.

13. Bachman, P.; Hjelm, R.D.; Buchwalter, W. Learning representations by maximizing mutual information across views. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019 ; pp. 15509–15519.
14. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
15. Li, C.L.; Sohn, K.; Yoon, J.; Pfister, T. Cutpaste: Self-supervised learning for anomaly detection and localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 9664–9674.
16. 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 (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
17. Kaggle. Airbus Ship Detection Challenge. Available online: https://www.kaggle.com/c/airbus-ship-detection/data (accessed on 31 July 2018).
18. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014; pp. 740–755.