remote sensing

MDPI

Article

On-Board Ship Detection in Micro-Nano Satellite Based on Deep Learning and COTS Component

Yuan Yao [1,2,3], Zhiguo Jiang [1,2,3], Haopeng Zhang [1,2,3,*] and Yu Zhou [4]

[1] Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China; yaoyuan@buaa.edu.cn (Y.Y.); jiangzg@buaa.edu.cn (Z.J.)
[2] Key Laboratory of Spacecraft Design Optimization and Dynamic Simulation Technologies, Ministry of Education, Beijing 100191, China
[3] Beijing Key Laboratory of Digital Media, Beijing 100191, China
[4] DFH Satellite Co., Ltd., Beijing 100094, China; zhouyu0408@163.com
[*] Correspondence: zhanghaopeng@buaa.edu.cn; Tel.: +86-10-8233-8061

Received: 1 March 2019; Accepted: 26 March 2019; Published: 29 March 2019

Abstract: Micro-nano satellites have provided a large amount of remote sensing images for many earth observation applications. However, the hysteresis of satellite-ground mutual communication of massive remote sensing images and the low efficiency of traditional information processing flow have become the bottlenecks for the further development of micro-nano satellites. To solve this problem, this paper proposes an on-board ship detection scheme based on deep learning and Commercial Off-The-Shelf (COTS) component, which can be used to achieve near real-time on-board processing by micro-nano satellite computing platform. The on-board ship detection algorithm based on deep learning consists of a feature extraction network, Region Proposal Network (RPN) with square anchors, Global Average Pooling (GAP), and Bigger-Left Non-Maximum Suppression (BL-NMS). With the help of high performance COTS components, the proposed scheme can extract target patches and valuable information from remote sensing images quickly and accurately. A ground demonstration and verification system is built to verify the feasibility and effectiveness of our scheme. Our method achieves the performance with 95.9% recall and 80.5% precision in our dataset. Experimental results show that the scheme has a good application prospect in micro-nano satellites with limited power and computing resources.

Keywords: on-board processing; ship detection; micro-nano satellite; deep learning; COTS component

1. Introduction

Owing to the characteristics of wide coverage, long duration, remote access, and high data collection volume, earth observation satellites play a key role in urban planning, traffic surveillance, geological exploration, disaster assessment, military reconnaissance, etc. Among these satellites, micro-nano satellites have the specific advantages of small size, low power consumption, short development cycle, suitable for networking and constellation, and low cost to complete many complex space missions. Therefore, micro-nano satellites have become the research hotspot in the fields of scientific research, national defense construction, and commercial application [1,2]. Global Maritime Domain Awareness (MDA) [3] is one of the important applications of micro-nano satellites, and the periodic full coverage data of micro-nano satellites form the data basis for the extraction of target information in a large area. Ships are the main carrier of maritime transport, therefore the accurate position of ship target is of great significance and value for MDA.

The traditional processing flow of ship detection from remote sensing images consists of mission planning, satellite imaging, satellite-earth data transmission, and ground processing. The ground processing scheme has time delay for ground users, and the satellite-earth data transmission system is under tremendous pressure. There are two bottlenecks for the further development of micro-nano satellites, i.e., the hysteresis of satellite-ground mutual communication of massive remote sensing images and the low efficiency of traditional information processing. On-board processing in micro-nano satellites is an effective way to improve the response speed to emergencies and provide immediate products for users. Figure 1 shows the comparison between on-board processing flow and the traditional ground processing flow. On-board image processing analyzes the remote sensing images directly on the satellite, and then only transmits the effective information to the ground receiver immediately. Compared with the traditional ground processing flow of ship detection, the advantages of on-board processing flow of ship detection are as follows: (1) the gigabyte level images are simplified to megabyte level target patches and information by on-board ship detection. It can reduce the compression, transmission and storage pressure for emergencies. (2) It simplifies the process of ground equipment and directly delivers the concerned information to users from the satellite. (3) Users can acquire information in near real-time while traditional processing flow needs several hours or days.

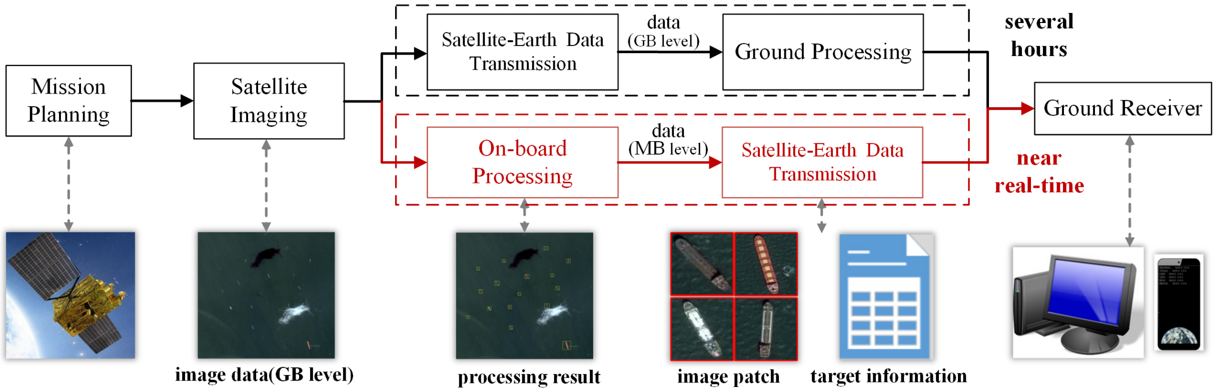

Figure 1. The comparison between ground processing and on-board processing. The black line is the traditional ground processing flow, and the red line is the on-board processing flow.

The core of on-board processing is the hardware architecture and the intelligent processing algorithm. Current on-board processing mainly applies the architecture of Field-Programmable Gate Array (FPGA) and Digital Signal Processor (DSP). Such architecture has limited computing capacity and the existing algorithms on the hardware architecture can only complete some simple tasks, such as image compression [4], ortho-rectification [5], point matching [6], and preprocessing [7]. With perceptually lossless compression methods [8], the satellites may have the ability to downlink more images to ground. However, the volume of compressed images is still too large for limited downlinking bandwidth even under high compression ratio, especially for urgent tasks and micro-nano platforms. As for the intelligent processing algorithm, there are two main problems for ship detection on micro-nano satellite. One is on how to use the limited computing resources on the satellite to detect ships from massive images. The on-board resources are severely limited (configuration, power consumption, computing capability, etc.) due to the small size and weight of micro-nano satellites. The other one is on how to efficiently detect ship targets in a large area under complex scenes (cloud, fog, wave, island interference, etc.).

With the development of Commercial Off-The-Shelf (COTS) component, it has become an important way to reduce the cost and shorten the development cycle of micro-nano satellites, which has the advantages of low cost, superior performance, high integration and easy availability. For the past few years, cheap, compact, low-power and high processing performance modules are emerged, such as Jetson TX1 and TX2 by NVIDIA Corporation. These modules are very adequate for on-board processing of micro-nano satellites.

In the past decade, some traditional methods have been proposed for ship detection [9–13]. Zhu et al. [9] applied shape and texture features to extract the region of interest, and then a semi-supervised hierarchical classification was adopted to eliminate the false alarms. Shi et al. [10] used anomaly detector and local shape feature to detect ships in a "coarse-to-fine" manner. Qi et al. [11] developed an unsupervised ship detection method based on saliency and S-HOG descriptor. Yang et al. [12] proposed a detection algorithm based on saliency segmentation and the local binary pattern descriptor combined with ship structure. Dong et al. [13] constructed a hierarchical model based on multi-scale saliency, rotation-invariant feature, and trainable Gaussian support vector machine classifier. Most of these methods are designed on image segmentation and hand-crafted features, which may have poor performance in complex background.

In recent years, deep learning has achieved great success in computer vision [14–17]. The development of deep learning technology makes it possible to detect ships with high accuracy and high processing speed. The on-board ship detection algorithm needs to detect as many targets as possible with limited resources and take both the detection accuracy and model size into consideration. However, most of the previous ship detection methods [18–26] detected ships on small image patches by high-power Graphic Processing Units (GPUs) on the ground, without considering the computing and storage resources for practical aerospace application.

To solve the problems above, this paper presents an on-board ship detection scheme based on COTS component and deep learning, which can be used to achieve near real-time on-board processing by micro-nano satellite computing platform. In addition, a ground demonstration and verification system is built to verify the feasibility and effectiveness of our scheme. Experimental results show that the proposed scheme has a good application prospect in micro-nano satellites with limited power and computing resources. The main contributions of this paper are three-fold.

1. We design an on-board ship detection scheme based on deep learning and COTS component. Different from the traditional ground processing flow, our scheme directly distributes the target patches and valuable information to ground users from micro-nano satellites, which can increase the response and processing speed, application ability, and the data utilization rate of micro-nano satellites.
2. We adopt the strategies of image cropping, square anchor, global average pooling, and BL-NMS to extract the target patches and information from remote sensing images. Compared to the previous ship detection methods, our method can achieve higher detection accuracy with a light weight model size, which meets the requirement of practical aerospace application with limited computational and storage resources.
3. We collect a well-annotated dataset from Google Earth, which includes the binary mask, bounding box, and squared bounding box of each ship. The performance of the proposed scheme tested by a ground demonstration and verification system demonstrates that our scheme has a promising application prospect in on-board intelligence processing.

The rest of this paper is organized as follows. Section 2 presents the details of our proposed ship detection algorithm. Section 3 introduces the construction of micro-nano satellite computing platform. Section 4 introduces the ground demonstration and verification system and experimental results are shown in Section 5. Section 6 discusses the practical application. Finally, Section 6 concludes this paper.

2. On-Board Ship Detection Based on Deep Learning

In the fields of image classification and object detection, the Convolution Neural Network (CNN) has made a great breakthrough in performance and efficiency. AlexNet [14] won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [27] 2012 by a large margin. After several years of development, the error rate of the winning scheme [28] in ImageNet classification competition reached 3.57% in 2015, while the error rate of human is 5.1%. In recent years, deep learning technology has been successfully applied in various fields. Girshick et al. first proposed an object detection framework based on Region-based Convolutional Networks (R-CNN) [29] in 2014, and then researchers proposed a series of object detection algorithms based on deep learning, such as Fast R-CNN [30], Faster R-CNN [16], You Only Look Once (YOLO) [31], Single Shot MultiBox Detector (SSD) [32], etc. These methods have surpassed the traditional algorithms in various object detection tasks.

Owing to the good performance of deep learning methods in object detection from natural scene images, an increasing number of methods based on deep learning are applied to remote sensing image processing. The deep learning algorithm has achieved good performance in the detection tasks of aircraft [33], oil tanks [34] and vehicles [35] from remote sensing images. Deep learning based ship detection methods have also been proposed [18–26]. These methods have achieved outstanding results in performance, but most of them deal with images on the ground without considering the computational and storage resources.

Compared with objects in natural scene image and other targets in remote sensing image, on-board ship detection from remote sensing images has the following difficulties:

- Due to the limitation of the Graphic Processing Unit (GPU) memory, the large-scale remote sensing image should be cropped into image patches. For images of different sizes, object detection methods from natural scene image are usually performed by rescaling the long side of input image to a certain length, such as 800 pixels. As the size of remote sensing image ranges from several thousands to tens of thousands of pixels, it is not feasible to use the remote sensing images with high down-sampling rate.
- The ship target varies from a few meters to a few hundred meters in length and has different directions. Since the ship target usually has a rigid body with symmetrical strip structure, it has a large aspect ratio.
- The ship target has large intraclass variations. Due to different requirements, the structure of ship body is designed into different shapes, which presents great differences in remote sensing images.
- The final detection bounding boxes need to be accurately located, and include the complete body of ship for further analysis in ground equipment. If the downlinked target patches miss ships or only include parts of ships, some valuable information will be lost.

To solve these difficulties, we adopt the strategies shown in Figure 2. We first crop the large scale remote sensing images into a series of 1024 × 1024 image patches, and then a lightweight model is employed to do feature extraction. The region proposals are generated by the Region Proposal Network (RPN) with square anchors. After Region of Interest (ROI) pooling layer and global average pooling layer, the bounding box regression and score of proposals are processed, followed by Bigger-Left Non-Maximum Suppression (BL-NMS) to remove redundant detection. Finally, the results in image patches are mapped into the input image coordinate to obtain the final result.

This paper focuses on ship detection on sea surface from optical remote sensing image. Compared with traditional synthetic aperture radar (SAR) images, optical images have the advantages of rich details and short revisit period. In addition, since the ships far away from the land has more important value than the inshore ship, our method mainly aims at offshore ships.

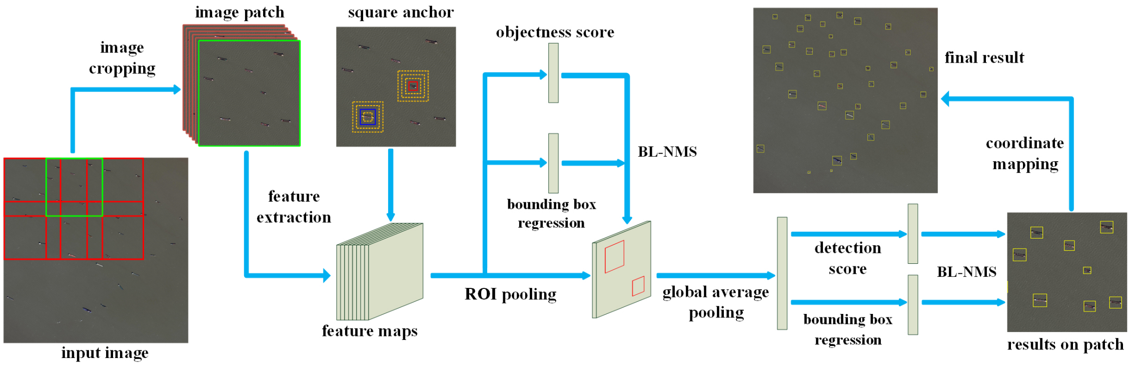

Figure 2. The process of on-board ship detection based on deep learning.

2.1. Image Cropping

Due to the large size of remote sensing images and the limitation of computing resources, we divide the large-scale remote sensing images into 1024 × 1024 image patches, which are used as the input images for training and testing. The longest existing ship is the Knock Nevis (458.45 m), which is an ultra large crude carrier from Singapore. In order to prevent the ship target from being cut off at the margin of patches, overlapping sampling is used during image cropping. The overlap step is set to 256 pixels in this paper.

2.2. Feature Extraction

Feature extraction is the basis of object detection and the quality of the features have great influence on follow-up tasks. With the higher performance requirements, the Zeiler and Fergus (ZF) [36] and the visual geometry group (VGG) [37] networks used in the original Faster R-CNN cannot meet the demand. Hence more CNN networks with better performance have been proposed in recent years, such as Residual Network (ResNet) [28], Dense Convolutional Network (DenseNet) [38], Squeeze-and-Excitation Network (SENet) [39], etc. Due to the limited on-board computing and storage resources, we adopt ResNet-50 as feature extraction network, which has a good performance and light weight.

2.3. Region Proposal Network with Square Anchor

RPN is used to extract the target proposals, which applies the anchor mechanism to ensure that targets of different positions and orientations can be covered by the proposals. The key parameters in RPN are the size and aspect ratio of anchors according to the training set. The length range of the ship target is very large, so both small anchor and large anchor should be taken into consideration. The length-to-width ratio of a ship target is generally around 3.0 to 7.2. Since ship targets have random orientations, the aspect ratio of bounding box for ship may range from 1:8 to 8:1. However, anchors with multiple aspect ratios will increase the amount of calculation, and some types of anchors with limited positive samples will generate a large number of false alarms. After determining the size and aspect ratio of anchors, there are two principles for choosing positive samples from anchors. (1) The anchor is assigned as a positive sample if its Intersection-over-Union (IoU) overlap with any ground-truth boxes is greater than 0.5, since the correct detection can only be calculated when the IoU overlap of the final detection result and ground-truth boxes is greater than 0.5. Reference [40] showed that the selection rules of training samples are closely related to the final detection results. (2) The anchor with highest IoU overlap with a ground-truth box is also determined to be a positive sample, in order to ensure that each target is covered by a positive sample at least.

According to these two principles and the characteristics of large aspect ratio and wide size distribution of ship target, we design square anchors. Figure 3 shows some positive samples of different types of anchors, where B and S represent original anchor and square anchor respectively, followed by the number of anchors. For the ships in training set, we use K-means to select four scales and choose three aspect ratios to adapt the rectangular target at the same time. As shown in Figure 3, for the target orientation approximately 45°, the square anchors (aspect ratio is 1) have better positive samples. However, for the target orientation 90° (or 0°), B-4 cannot choose appropriate samples, and B-12 can select the positive samples close to the target. Compared with B-4 and B-12, positive samples of good quality can be selected by S-4 under various circumstances, which will provide good support for subsequent networks.

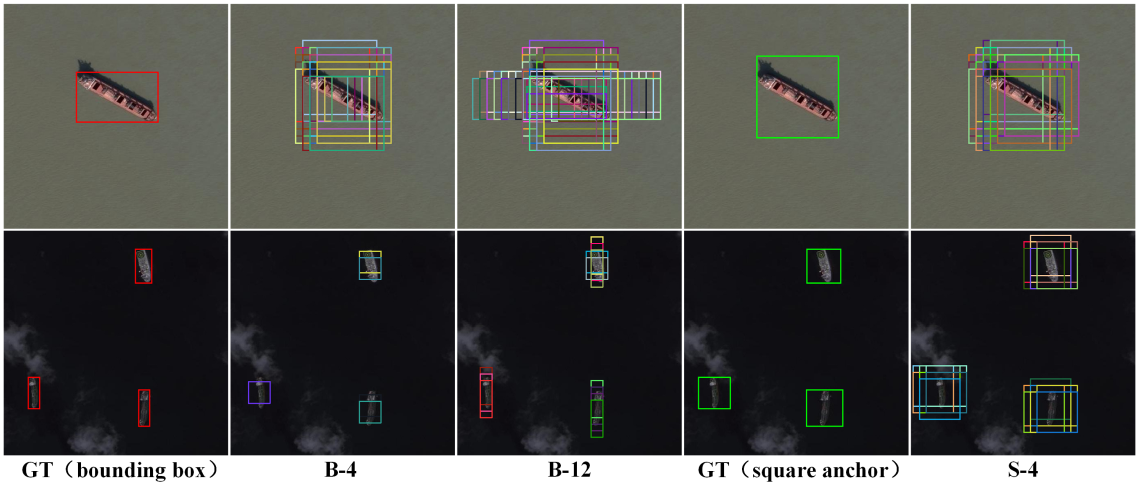

Figure 3. Some positive examples of different types of anchors. B and S represent original anchor and square anchor, respectively, followed by the number of anchors.

We form a multi-task loss function for training RPN, which is defined as follows

$$L\left(\{p_i\},\{t_i\}\right) = \frac{\lambda_1}{N_{cls}}\sum_i L_{cls}(p_i,{p_i}^*) + \frac{\lambda_2}{N_{det}}\sum_i {p_i}^* L_{det}(t_i,{t_i}^*) \quad (1)$$

where i is the index number of anchors, p_i is the probability that the image patch of the anchor i is ship target, ${p_i}^*$ is the ground truth label (1 represents that the anchor is positive sample, and 0 means that the anchor is non-ship target), t_i is the predicted position vector of candidate regions, and ${t_i}^*$ is the ground truth for the position vector of the candidate region. The cross entropy loss function L_{cls} is adopted to represent the classification loss of candidate regions. L_{det} is smooth L1 loss, which is counted for bounding box regression. The balance factors λ_1 and λ_2 are set as 1 and 10 respectively. In order to normalize each loss, N_{cls} is the number of candidate regions to be classified, and N_{det} equals the number of positive samples selected from anchors.

The regression task of RPN is to regress four offset values, $t = (t_x, t_y, t_w, t_h)$, from an anchor to the corresponding ground truth. We adopt the parameterization for t given in Reference [16], where t is defined as

$$\begin{aligned} t_x &= (x - x_a)/w_a, t_y = (y - y_a)/h_a \\ t_w &= log(w - w_a), t_h = log(h - h_a) \\ t_x^* &= (x^* - x_a)/w_a, t_y^* = (y^* - y_a)/h_a \\ t_w^* &= log(w^* - w_a), t_h^* = log(h^* - h_a) \end{aligned} \quad (2)$$

where x, y, w, and h denote the bounding box's center coordinates, width and height, and (x, y, w, h), (x_a, y_a, w_a, h_a), and (x^*, y^*, w^*, h^*) are the parameters of the predicted box, anchor, and ground-truth respectively.

2.4. Global Average Pooling

The parameters of Full Connection (FC) layers account for about 80% of the parameters of the whole network, resulting in a large model, which is not suitable for on-board applications. Therefore, the method of GAP [41] is adopted after the ROI Pooling layer to reduce the parameters of the model. One advantage of global average pooling layer over the fully connected layers is that it is more native to the convolution structure and another advantage is that there is no parameter to optimize in the global average pooling, thus over-fitting is avoided at this layer [41].

2.5. Bigger-Left Non-Maximum Suppression

The output of the trained network will produce a large number of overlapping detection boxes, and the traditional algorithms usually adopt the NMS method to remove the redundancy. The NMS method is used to calculate the IoU overlap of two boxes according to the score of the detection boxes one by one. If the IoU overlap is greater than the threshold, the detection box with lower score will be removed. If the IoU overlap is less than the threshold, the two detection boxes will be retained. As shown in Figure 4, the NMS method can remove a lot of overlapping boxes. If the box with high score cover the ship target incompletely, it will lead to the target patch only contains parts of ships, which influences the further recognition and analysis for ground users.

To reduce incomplete target patches, we use the method of BL-NMS. Algorithm 1 illustrates the process of BL-NMS in detail. Firstly, the traditional NMS method is employed to calculate the IoU overlap of two detection boxes. Then if the IoU overlap is greater than the threshold, the detection box with high score will be updated to a larger detection box that can fully contain the two detection boxes. Finally the detection boxes can contain more target areas to ensure that the target will not be truncated when all detection boxes are refined by BL-NMS. Figure 4 shows the results of different NMS methods. The target patches by BL-NMS have a better coverage of ships than the target patches by NMS. In this paper, we choose 0.3 as the threshold of BL-NMS.

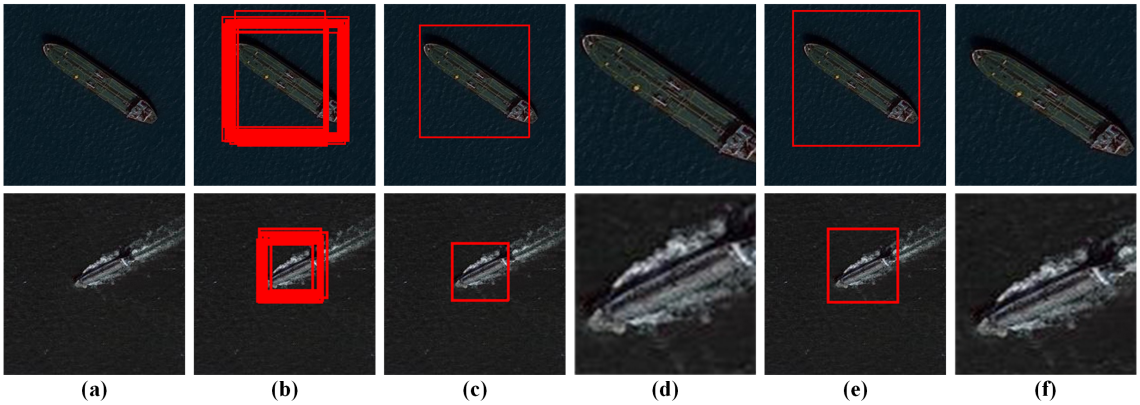

Figure 4. The results of different Non-Maximum Suppression (NMS) methods. (**a**) The small slice of input image. (**b**) The detection result of the network with many overlapping boxes. (**c**) The detection result of traditional NMS. (**d**) The target patch extracted by traditional NMS. (**e**) The detection result of Bigger-Left Non-Maximum Suppression (BL-NMS). (**f**) The target patch extracted by BL-NMS.

Algorithm 1 Bigger-Left Non-Maximum Suppression (BL-NMS)

```
Input:
    $\mathbf{B} = \{b_1,...,b_N\}, \mathbf{S} = \{s_1,...,s_N\}, \mathbf{B}_{final} = \{\}, T,$
    $\mathbf{B}$ is the list of detection boxes before BL-NMS, and S is the list of detection scores of each boxes.
    $\mathbf{B}_{final}$ is the list of detection boxes after BL-NMS, and T is the threshold of BL-NMS.
    N is the number of detection boxes.
    $b_i = (x_1, y_1, x_2, y_2)$, where $(x_1, y_1)$ and $(x_2, y_2)$ denote the positions for top-left and bottom-right points of detection boxes.
while $\mathbf{B} \neq \emptyset$ do
    $ind = argmax(\mathbf{S})$
    $b_{buf} = b_{ind}, b_{bigger} = b_{ind}$
    $\mathbf{B} = \mathbf{B} - b_{buf}$
    for $b_i$ in $\mathbf{B}$ do
        if IoU($b_{buf}, b_i$) $\geq$ T then
            $\mathbf{B} = \mathbf{B} - b_i, \mathbf{S} = \mathbf{S} - s_i$
            $b_{bigger} = (min(b_{bigger}(x_1), b_i(x_1)), min(b_{bigger}(y_1), b_i(y_1)),$
                    $max(b_{bigger}(x_2), b_i(x_2)), max(b_{bigger}(y_2), b_i(y_2)))$
            $\mathbf{B}_{final} = \mathbf{B}_{final} \bigcup b_{bigger}$
        end if
    end for
end while
return $\mathbf{B}_{final}, \mathbf{S}$
```

3. Construction of Computing Platform in Micro-Nano Satellite

3.1. The Integrated Information Flow of Platform and Payload

Micro-nano satellites refer to small satellites weighing less than 100 kg, such as SkySat, Flock-1 and so on. In 2017, 467 spacecrafts were launched around the world, of which 321 were micro-nano satellites (68.7%). A large amount of data for earth observation can be generated by micro-nano satellites, and these data will be downlinked when the satellites fly over the ground stations, which is not time-sensitive. Besides, the limited bandwidth of satellite-earth data transmission leads to many valuable images not downlinked to ground. This situation significantly limits the application ability of micro-nano satellites.

To alleviate the pressure of data transmission, the remote sensing image can be simplified by some on-board processing on the satellites. The small size of the configuration brings a variety of resource limitations on micro-nano satellites, such as power consumption, energy, computing power and so on. Therefore, it is difficult to achieve advanced tasks. At present, traditional on-board processing tasks are mainly focused on image preprocessing, mass data storage and format conversion, data compression, automatic analysis of original data and target feature extraction, etc. With the improvement of spatial resolution of remote sensing image and the sharp increase of data volume, it not only forms data basis for the higher information processing on the satellite, but also brings challenges to the performance, flexibility, power consumption and other aspects of the on-board processing technology.

The integrated information flow of platform and payload is an effective method to improve the capacity of imaging and processing for micro-nano satellites. The technology has the advantages of improving the location precision for remote sensing image, reducing satellite quality, and reducing construction cost. We use heterogeneous backup of System on Chip (SoC) and Graphic Processing Unit (GPU) to meet the demand of the integration of information flow. SoC is utilized to achieve satellite platform and payload management and control with high reliability, including remote metering, remote control, attitude and orbit control, time management, payload management, recovery processing, etc. GPU with high-performance data processing capability is applied to do on-orbit real-time remote sensing image processing. Besides, the satellite can use the CPU embedded in the GPU processing module as the backup of SoC, not only ensure that satellite platform safe and reliable, but also meet the requirements of on-board high performance computing.

Figure 5 shows the hardware architecture of the integrated information flow. The information flow is interacted through CAN bus and satellite affairs, measurement, navigation and other functions are controlled through integrated information flow unit. The imaging of remote sensing payload unit is adjusted by integrated energy flow unit and attitude control unit, and then remote sensing images are transmitted to GPU and satellite-ground communication unit through camera link respectively. Target patches and information processed by GPU are transmitted to the satellite-ground communication unit through LVDS and then satellite-ground communication unit transmits these data to the ground receiver by antenna. The micro-nano satellite needs to choose the type of data (images or patches and information) with the limitation of the downlink capability.

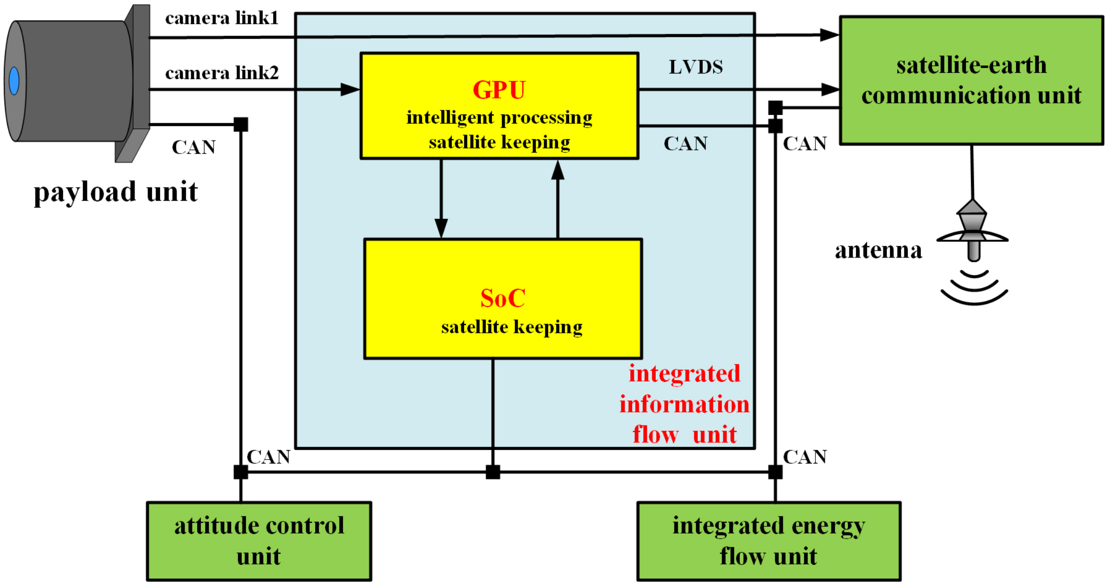

Figure 5. Hardware architecture of the integrated information flow.

3.2. The Commercial GPU in Integrated Information Flow Unit

Compared with traditional large satellites, micro-nano satellites do not need a long lifetime. The disadvantage of a short lifetime can be compensated by networking and constellation. Therefore, the COTS components with the characteristics of low cost, superior performance, high integration and easy access are very suitable for on-board processing in micro-nano satellites.

Embedded component Jetson TX2 by NVIDIA Corporation, which is a development board integrated four cores ARM Cortex-A57 CPU and 256 cores Pasal GPU, can achieve a trillion times floating point number operations per second on the Linux system. Jetson TX2 is technically designed for computer vision, deep learning and other fields with the same size of a credit card. Low cost ($599), low power consumption (less than 7.5 W), small size (50 mm × 87 mm), and light weight (85 g) are all suitable for on-board processing on micro-nano satellites. Jetson TX2 has the software environment for the development tools of deep learning, so the code and model of proposed method can be easily ported into Jetson TX2.

3.3. Intelligent Application

A variety of advanced on-board processing tasks can be realized through the high-performance processing platform, which can simplify massive remote sensing images into valuable information and greatly improve the data utilization rate. We receive near real-time downlinked data from satellite through the data service operation platform and analyze it in some intelligent application (APP). The received data will be graphically displayed by the graphical display module for the convenience of users. Through the monitoring and management software, the ground users can realize the monitoring of the data transmitted from the satellite and the communication supervision of the ground instructions uploaded to the satellite. Besides, the algorithm model and code of the new task can be uploaded to the information flow unit of the satellite through the satellite upload channel, so as to accomplish new applications.

4. The Composition of Ground Demonstration and Verification System

In order to verify the feasibility and effectiveness of the proposed scheme, we set up a ground demonstration and verification system. The operation process is shown in Figure 6.

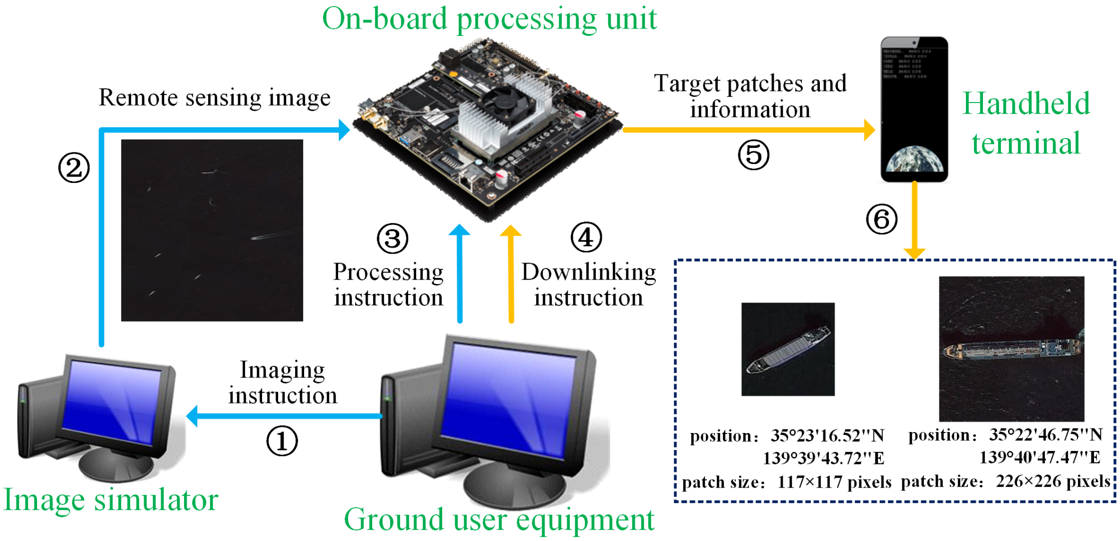

Figure 6. The operation process of the ground demonstration and verification system.

The ground users first determine the observation area according to the requirements, and then send an imaging instruction to the image simulator to simulate the function of payload imaging. After the image simulator selects the image of the relevant area, the image is transmitted to the on-board intelligent processing unit. When the imaging is completed, the ground users send a processing instruction to the on-board processing unit. Then, the on-board processing unit uses the proposed method to detect ship target from the image. The target slices and information are extracted from the large-scale image in one minute. After the on-board processing, the ground users send a data transmission instruction to the on-board intelligent processing unit to transmit the target patches and information to the ground handheld terminal. Finally, the users can check the target patches and information through intelligent APP. The following is the introduction of each device.

4.1. Ground User Equipment

According to the requirements of different tasks, the ground user equipment completes the functions of intelligent processing application scene configuration, parameter configuration and instruction control with the remote sensing satellite. The application scene configuration is to select the specific algorithm, and parameter configuration includes satellite data information, the selection of intelligent algorithm model and the data transmission mode. Instruction control includes image simulation, data generation, intelligent processing and product generation instruction.

4.2. Image Simulator

The image simulator is used to simulate payload imaging and its main function is to provide remote sensing images for the entire system. It includes two different methods to generate the required images. One is to directly use the real remote sensing data as input data, and the other one is to combine the task requirements and the image synthesis method in Reference [42] to generate simulation images for application scenes. The latter method is applied to deal with the unusual scenes and enhance the generalization of the proposed algorithm.

4.3. On-Board Processing Unit

The processing unit on the satellite is the core of the ground demonstration and verification system. This module uses Jetson TX2 combined with deep learning algorithm to simplify a large amount of remote sensing data into the information required by users, which improves the utilization rate of data and reduces the pressure of data transmission. The code and trained model of proposed scheme need to be stored in the intelligent processing unit before the operation of the system. Since the model and code can be updated or replaced in the information flow unit through the upload channel, the whole framework can also be applied to other high-level tasks, such as other target detection, change detection, terrain classification, etc., which can greatly improve the application ability of micro-nano satellites.

4.4. Ground Handheld Terminal

The ground handheld terminal mainly displays the processing results obtained by the on-board processing unit on the satellite. In the form of APP, the target patches and information outputted by the on-board processing unit are displayed on the handheld terminal to verify that the whole system is working properly. By receiving target patches and information, the user can further analyze the target's attributes and higher-level tasks, such as type classification, size measurement, and analysis of motion state.

5. Experimental Results

5.1. Dataset

In order to verify the effectiveness of the scheme in this paper, we established a dataset collected from Google Earth for the lack of publicly available datasets for on-board ship detection. The dataset contains 95 images under different imaging conditions with the resolution of 1 m. Training set includes 65 images and the remainder are used as test images. The image size of our dataset ranges from 1500 × 1500 to 10,000 × 10,000 pixels. In total, there are 1397 ships in the training set and 435 ships in the test set. We rotated the training image 36 times (per 10 degrees) to generate different directions, and cropped 12,965 training samples with the size of 1024 × 1024 pixels from the augmented dataset. After data augmentation, 42,308 ships of different types, sizes and directions were used for training. We manually marked the ground truth of all ship targets to obtain the performance of our scheme. As shown in Figure 7, the mask of ship target was marked firstly, and then the minimum bounding rectangle of the connected area was calculated. Finally, the short side of the minimum bounding rectangle was stretched to be the same as the long side to ensure the final bounding box was square. At the same time, we only focus on ships larger than 20 pixels in length, and small ships with long wake are not labeled as positive samples, since their exact location is difficult for human eyes to distinguish. Our dataset includes images under different conditions, such as calm sea, clutter sea, moving target, cloud, and fog interference, which is appropriate for simulating the practical application.

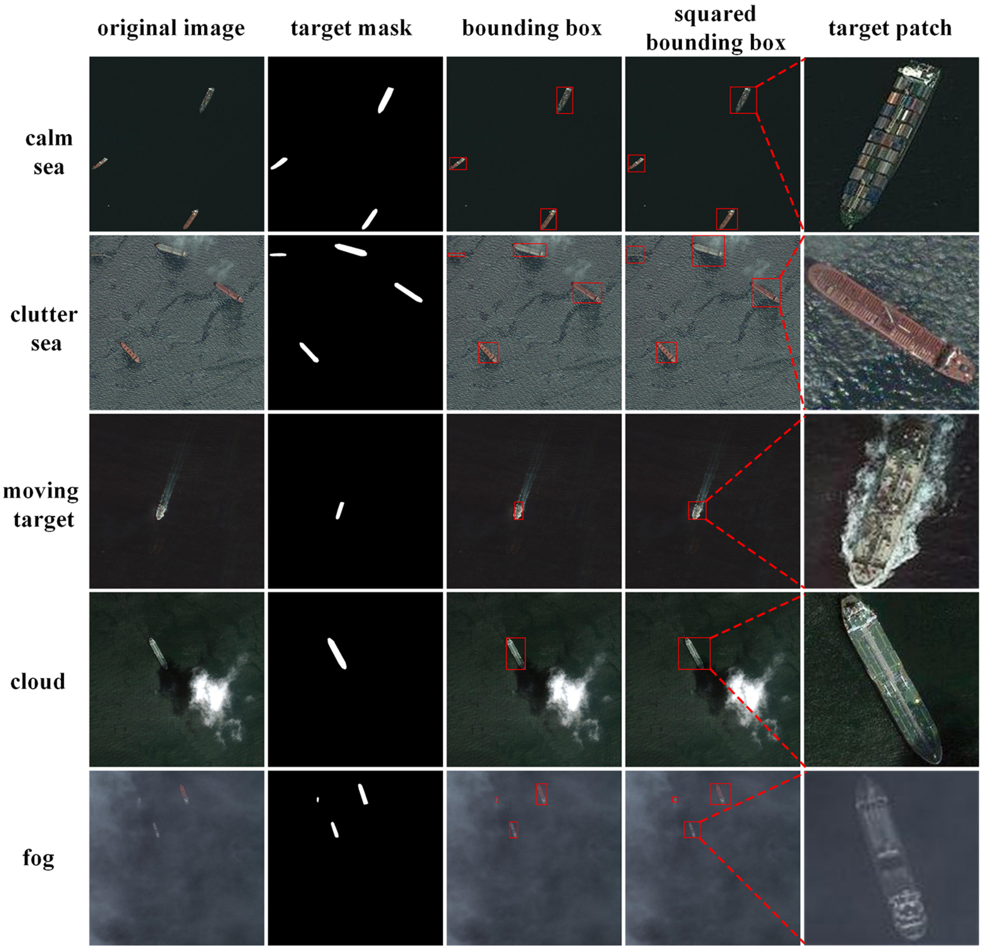

Figure 7. Some image patches under different conditions in our dataset.

5.2. Implementation Details

All experiments were implemented by MXNet [43], which is a scalable deep learning framework. We used the pretraining model ResNet-50 to initialize the network. All models were trained for eight epoches on our dateset, with a learning rate of 0.001 for the first six epoches, and 0.0001 for the next two epoches. The optimizer is Stochastic Gradient Descent (SGD) and weight decay and momentum were 0.0005 and 0.9, respectively. The programming language of the algorithm is Python. Considering the capacity of the GPU in Jeston TX2 and training efficiency, we trained the CNN models on a PC with one Nvidia GTX 1080Ti to accelerate. Then the well-trained models by PC were ported into Jeston TX2.

Table 1 shows the statistics of different types of anchors. S-4 has more positive samples and bigger average IoU of positive samples than B-4 and B-12. The number of anchors in B-12 has three times as many as S-4, but the anchors of S-4 still has better quality both in quantity and IoU overlap with ground truth.

Table 1. The statistics of different types of anchors. B and S represent original anchor and square anchor respectively, followed by the number of anchors.

Anchor Type	Scale	Aspect Ratios	Positive Samples/Image(s)	Positive Samples (IoU > 0.5)/Image(s)	Average IoU of Positive Samples
B-4	1.42, 3.48, 6.40, 10.50	1	13.60	7.32	0.579
B-12	1.42, 3.48, 6.40, 10.50	0.33, 1, 3	14.99	14.08	0.578
S-4	2.06, 5.29, 9.40, 15.19	1	23.83	22.88	0.604

5.3. Processing Performance

This subsection tests the effectiveness of the proposed method. To quantitatively evaluate the performance of different frameworks, we use recall and precision as evaluation metrics of ship detection. We provide the definitions of precision and recall rate as

$$Recall = \frac{Number\ of\ ship\ targets\ detected}{Number\ of\ ship\ targets\ in\ the\ dataset} \tag{3}$$

$$Precision = \frac{Number\ of\ ship\ targets\ detected}{Number\ of\ detection\ boxes} \tag{4}$$

Table 2 summarizes the experimental results of different methods. We also conducted ablation experiments on our dataset. The performance of S-HOG [11] is poor, because our dataset is complex and this unsupervised approach based on saliency and the S-HOG descriptor needs many empirical parameters. YOLOv3-tiny [44] is an improved version of YOLO, and it has the smallest model size but suffers the sacrifice of detection accuracy. Other methods in Table 2 are under our scheme. The method in [21] is not suitable for practical application due to the model size of 521.5MB. Mobilenet [45] has a small model size, which is an efficient convolutional neural networks for mobile vision applications. Mobilenet uses depthwise separable convolution to reduce the parameters and the recall has a little decline at the same time. The decline of recall is unacceptable for on-board processing since the missing targets will not be transmitted to the ground receiver. The trade-off between performance and model size is always a difficult problem. For on-board ship detection, we think that the performance is more important than model size. Finally, we choose Resnet-50 as our baseline network.

Table 2. Comparisons of different methods.

Method	Anchor type	FC/GAP	Recall	Precision	Model Size (MB)
S-HOG [11]	–	–	0.623	0.587	–
YOLOv3-tiny [44]	S-4	FC	0.807	0.771	30.1
Method in [21]	S-4	FC	0.954	0.740	521.5
Mobilenet [45]	S-4	FC	0.876	0.836	64.82
Resnet-50-1	S-4	FC	0.920	0.774	250.81
Resnet-50-2	B-4	GAP	0.933	0.707	108.03
Resnet-50-3	B-12	GAP	0.959	0.757	108.12
Resnet-50-4	S-4	GAP	0.979	0.816	108.03

Resnet-50-1 and Resnet-50-4 were designed to evaluate the effect of the replacement of FC layers by GAP layer. The size of GAP layer depends on the output of the network and it is 7 × 7 in Resnet-50-4. Resnet-50-4 has a performance improvement over Resnet-50-1 by a large margin and reduces the model size from 250.81 MB to 108.03 MB, which is very practical for on-board processing. The advantage of using the strategy of square anchors can also be shown in Table 2. Resnet-50-2, Resnet-50-3, and Resnet-50-4 are designed to evaluate different anchor types. S-4 has a better performance than B-4 and B-12 since it can provide higher quality positive samples. This demonstrates that using square anchors has a good effect on the scheme. Overall, Resnet-50-4 gets the best performance with 97.9% recall and 81.6% precision.

Table 3 is the performance comparisons by different NMS methods. BL-NMS method has a little performance decrement over the traditional NMS, since BL-NMS enlarges some detection boxes for small targets to a bigger size, which has the IoU overlap less than 0.5 with ground truth. To evaluate the quality of the ship patches generated by our method, strict recall and strict precision are used as the evaluation metrics. The ground users need to scan the whole parts of ships, hence our detection boxes should cover the ship completely, which means that the minimum bounding rectangle of each ship needs to be contained within the detection boxes. A detection box is counted as true positive in strict recall and strict precision, only when its IoU overlap with ground truth is greater than 0.5 and it also contains the ground truth. The stricter metrics lead to a large performance decrement for traditional NMS while it has only a little impact on BL-NMS. The results demonstrate the effectiveness of BL-NMS in improving the quality of the ship patches generated by the on-board processing method.

Table 3. Performance comparisons using different NMS methods and NMS thresholds.

Method	Recall	Precision	Strict Recall	Strict Precision
NMS (0.3)	0.979	0.816	0.228	0.190
BL-NMS (0.3)	0.959	0.805	0.920	0.772
BL-NMS (0.5)	0.979	0.747	0.929	0.710
BL-NMS (0.7)	0.993	0.502	0.933	0.472

We also evaluated the effect of the threshold of BL-NMS. The smaller threshold of BL-NMS can reduce more redundant detection boxes, which can achieve a higher precision. For on-board ship detection task, all the ship patches by the detection boxes are downlinked to the ground receiver, therefore the users can still scan the ships treated as false alarms. Finally, we choose 0.3 as the threshold of BL-NMS. If the satellite has a better transmittability, we can increase the threshold of BL-NMS to downlink more target patches.

5.4. Hardware Test

As shown in Table 4, we tested the models both on a PC with a GTX 1080Ti and Jetson TX2. The results certify that the detection performance of the two hardware environments are the same, which ensures that we can train the models on the PC and then copy the models to Jetson TX2. PC with a GTX 1080Ti has a high speed for training and testing, thus we trained all the models on the PC.

Table 4. Comparisons of different hardware environments.

Hardware Environment	Precision	Recall	Training Time/Samples	Processing Time (1 K × 1 K)
PC (GTX 1080Ti)	0.959	0.805	0.4 s	0.08 s
Jetson TX2	0.959	0.805	7.68 s	1.25 s

In order to test the feasibility of using Jetson TX2 on micro-nano satellites, we summarize some indexes of Jetson TX2 in laboratory. Table 5 provides the results and all the indexes are suitable for micro-nano satellites. The characteristics of cheap cost, small size and light weight mean that we can even layout several Jetson TX2 on micro-nano satellites. The standby power and peak power of Jetson TX2 are 3∼5 W and 15 W, respectively, which is not a great burden for energy system in micro-nano satellites. Besides, the temperature of Jetson TX2 is usually less than 45 °C, which is not a challenge for thermal control system. The indexes above demonstrate that Jetson TX2 can become a part of computing platform in micro-nano satellites. The ship detection method based on deep learning can be operated on Jetson TX2 due to its computing capability of 1 TFLOPS. The final model size of our method is 108.03 MB and the average processing time of 1 K × 1 K image patches is 1.25 s. These two indexes meet the requirements of on-board ship detection on micro-nano satellites for storage resource and processing time.

Table 5. The indexes of Jetson TX2 tested in laboratory.

Indexes	Test Results in Laboratory
Cost	$599
Weight	85 g
Size	50 mm × 87 mm
Power	3∼5 W (peak: 15 W)
Temperature	<45 °C
Computing capability	1 TFLOPS
Model size	108.03 MB
Processing time	1.25 s (1 K × 1 K)

6. Discussion

Many experiments in Section 5 verify the validity of the proposed scheme. We can use the Jetson TX2 as a part of integrated information flow unit in micro-nano satellite and achieve on-board ship detection algorithm through the unit. Figure 8 illustrates an example for the result of our scheme. The red boxes are the detection results of our ship detection method and the white boxes are the target patches extracted by our network. The two numbers are the image coordinates of the central point for each ships and they will be converted into geographic coordinates through coordinate conversion. The target patches and geographic coordinates are transmitted to the ground receivers. As shown in Figure 8, our scheme has good performance both in detection and patch quality and the results verify the effectiveness of strategies in our scheme. The file size of the image is 90.6 MB while the target patches and information are 810 KB in total. The reduction in file size is the biggest advantage of on-board ship detection, which helps to alleviate the pressure of satellite-ground mutual transmission and ground receivers.

For some emergencies, the ground users can control several micro-nano satellites in networking or constellation to obtain the raw remote sensing images over the areas focused on. Then the on-board ship detection algorithm is applied to extract the target patches and information from these raw remote sensing images. Finally, the target patches and information are transmitted to the ground user by relay satellites or other methods in a short time, which can improve the response speed to emergencies. Some perceptually lossless compression methods can be used to achieve image compression for target patches, which can further reduce the file size of target patches. The ground users can make better decisions to deal with emergencies by these valuable data. For other cases, the micro-nano satellites downlink the raw remote sensing images to the ground when the satellites fly over the ground stations. The raw data can be used for other purposes.

To achieve other on-board tasks, the users should store the programs and models in the integrated information flow unit on the ground. During the micro-nano satellites operating in orbit, the programs and models for new tasks can also be updated or uploaded to the satellite through the satellite upload channel. The micro-nano satellites can achieve many on-board intelligent processing tasks by the computing platform proposed in our scheme, which greatly improves the application ability of micro-nano satellites.

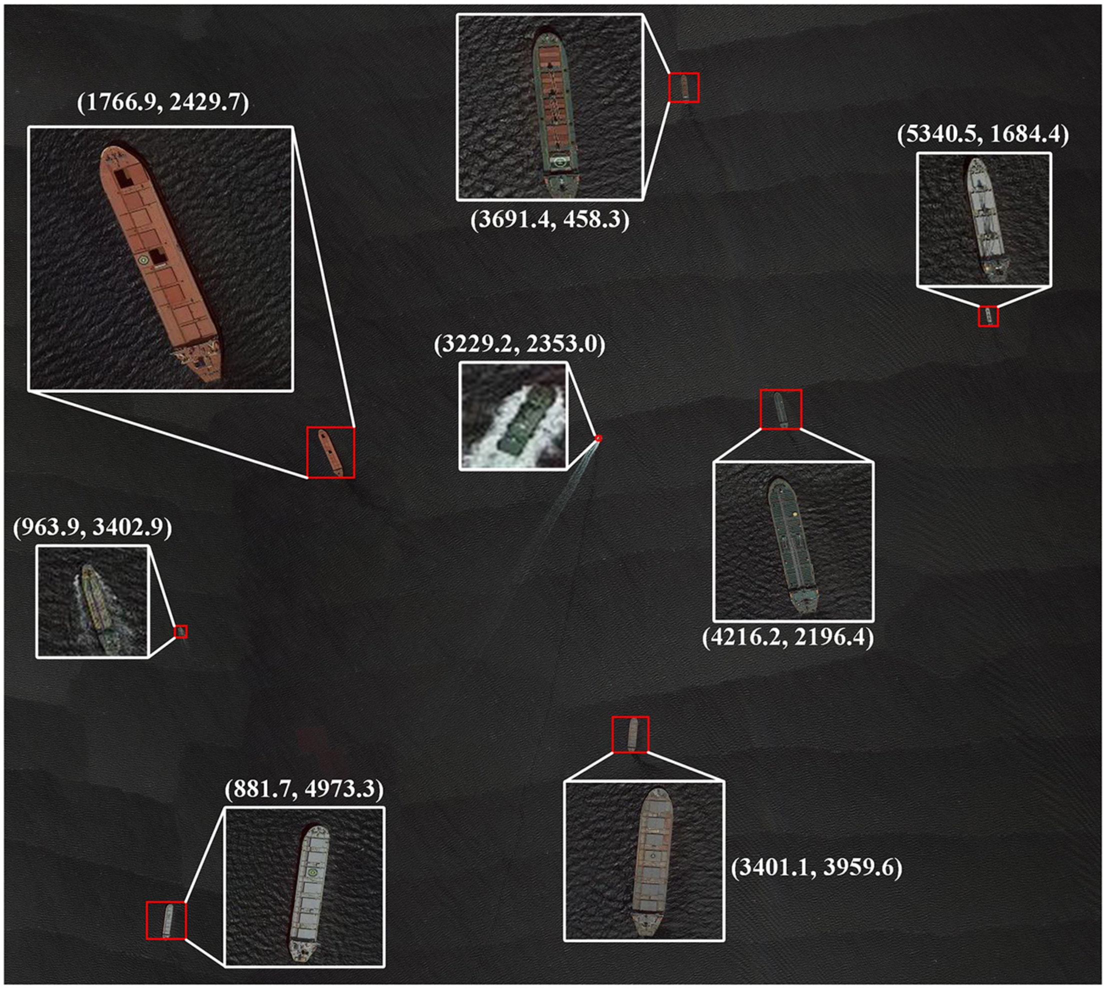

Figure 8. The result of the proposed scheme for one image in our dataset. The red boxes are the detection results of our ship detection method and the white boxes are the target patches extracted by our network. The two numbers are the image coordinates of the central point for each ships.

7. Conclusions

In this paper, we proposed an on-board ship detection scheme based on deep learning and a COTS component, which can be used to achieve near real-time on-board processing by micro-nano satellite computing platform. Many novel strategies were designed for on-board processing. Square anchors was used to enhance the quality of proposals and global average pooling layer replaced the full connection layers in the network to reduce the model size. In addition, bigger-left non-maximum suppression method was adopted to ensure the integrity of targets patches. We also constructed a computing platform in a micro-nano satellite and a ground demonstration and verification system to verify the feasibility and effectiveness of our scheme. Our method achieved the performance with 95.9% recall and 80.5% precision in our dataset and the target patches downlinked by our scheme have a better quality. Experiments showed that the scheme has a good application prospect for micro-nano satellites. In our future work, a more specific network will be designed to further reduce the model size and improve the computational efficiency. Another important work is the design of some protective measures for the reliability of COTS component.

Author Contributions: Y.Y., Z.J., H.Z., and Y.Z. conceived of this study. Y.Y. performed the experiments and wrote the manuscript. Z.J. and H.Z. supervised the work and revised the manuscript. Y.Z. contributed with field experience in micro-nano satellite and on-board processing.

Funding: This work was supported in part by the National Key Research and Development Program of China (2016YFB0501300 and 2016YFB0501302), the National Natural Science Foundation of China (Grant Nos. 61771031, 61501009 and 61371134), and the Fundamental Research Funds for the Central Universities.

Acknowledgments: The authors would like to thank all the colleagues in the lab, who generously collected the dataset with ground truth. Besides, we would also like to thank the editor and the three anonymous reviewers for their important comments and helpful suggestions.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Michael, Y.; Lensky, I.M.; Brenner, S.; Tchetchik, A.; Tessler, N.; Helman, D. Economic Assessment of Fire Damage to Urban Forest in the Wildland—Urban Interface Using Planet Satellites Constellation Images. *Remote Sens.* **2018**, *10*, 1479. [CrossRef]
2. Jain, M.; Srivastava, A.K.; Balwinder-Singh; Joon, R.K.; McDonald, A.; Royal, K.; Lisaius, M.C.; Lobell, D.B. Mapping Smallholder Wheat Yields and Sowing Dates Using Micro-Satellite Data. *Remote Sens.* **2016**, *8*, 860. [CrossRef]
3. Kanjir, U.; Greidanus, H.; Oštir, K. Vessel detection and classification from spaceborne optical images: A literature survey. *Remote Sens. Environ.* **2018**, *207*, 1–26. [CrossRef] [PubMed]
4. Guerra, R.; Barrios, Y.; Díaz, M.; Santos, L.; López, S.; Sarmiento, R. A New Algorithm for the On-Board Compression of Hyperspectral Images. *Remote Sens.* **2018**, *10*, 428. [CrossRef]
5. Zhou, G.; Zhang, R.; Liu, N.; Huang, J.; Zhou, X. On-Board Ortho-Rectification for Images Based on an FPGA. *Remote Sens.* **2017**, *9*, 874. [CrossRef]
6. Huang, J.; Zhou, G. On-Board Detection and Matching of Feature Points. *Remote Sens.* **2017**, *9*, 601. [CrossRef]
7. Qi, B.; Shi, H.; Zhuang, Y.; Chen, H.; Chen, L. On-Board, Real-Time Preprocessing System for Optical Remote-Sensing Imagery. *Sensors* **2018**, *18*, 1328. [CrossRef]
8. Kwan, C.; Larkin, J.; Budavari, B.; Chou, B. Compression Algorithm Selection for Multispectral Mastcam Images. *Signal Image Process. Int. J. (SIPIJ)* **2019**, *10*, 1–14. [CrossRef]
9. Zhu, C.; Zhou, H.; Wang, R.; Guo, J. A Novel Hierarchical Method of Ship Detection from Spaceborne Optical Image Based on Shape and Texture Features. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 3446–3456. [CrossRef]
10. Shi, Z.; Yu, X.; Jiang, Z.; Li, B. Ship Detection in High-Resolution Optical Imagery Based on Anomaly Detector and Local Shape Feature. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 4511–4523. [CrossRef]
11. Qi, S.; Ma, J.; Lin, J.; Li, Y.; Tian, J. Unsupervised Ship Detection Based on Saliency and S-HOG Descriptor From Optical Satellite Images. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 1451–1455. [CrossRef]
12. Yang, F.; Xu, Q.; Li, B. Ship Detection From Optical Satellite Images Based on Saliency Segmentation and Structure-LBP Feature. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 602–606. [CrossRef]
13. Dong, C.; Liu, J.; Xu, F. Ship Detection in Optical Remote Sensing Images Based on Saliency and a Rotation-Invariant Descriptor. *Remote Sens.* **2018**, *10*, 400. [CrossRef]
14. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
15. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. [CrossRef]
16. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef]

17. Kwan, C.; Chou, B.; Kwan, L.Y.M. A Comparative Study of Conventional and Deep Learning Target Tracking Algorithms for Low Quality Videos. In *International Symposium on Neural Networks*; Springer: Berlin, Germany, 2018; pp. 521–531.
18. Yao, Y.; Jiang, Z.; Zhang, H.; Zhao, D.; Cai, B. Ship detection in optical remote sensing images based on deep convolutional neural networks. *J. Appl. Remote Sens.* **2017**, *11*, 042611. [CrossRef]
19. Tang, J.; Deng, C.; Huang, G.; Zhao, B. Compressed-Domain Ship Detection on Spaceborne Optical Image Using Deep Neural Network and Extreme Learning Machine. *IEEE Transa. Geosci. Remote Sens.* **2015**, *53*, 1174–1185. [CrossRef]
20. Zou, Z.; Shi, Z. Ship Detection in Spaceborne Optical Image With SVD Networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 5832–5845. [CrossRef]
21. Han, X.; Zhong, Y.; Zhang, L. An Efficient and Robust Integrated Geospatial Object Detection Framework for High Spatial Resolution Remote Sensing Imagery. *Remote Sens.* **2017**, *9*, 666. [CrossRef]
22. Yang, X.; Sun, H.; Fu, K.; Yang, J.; Sun, X.; Yan, M.; Guo, Z. Automatic Ship Detection in Remote Sensing Images from Google Earth of Complex Scenes Based on Multiscale Rotation Dense Feature Pyramid Networks. *Remote Sens.* **2018**, *10*, 132. [CrossRef]
23. Yang, X.; Sun, H.; Sun, X.; Yan, M.; Guo, Z.; Fu, K. Position Detection and Direction Prediction for Arbitrary-Oriented Ships via Multitask Rotation Region Convolutional Neural Network. *IEEE Access* **2018**, *6*, 50839–50849. [CrossRef]
24. Liu, W.; Ma, L.; Chen, H. Arbitrary-Oriented Ship Detection Framework in Optical Remote-Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 937–941. [CrossRef]
25. Zou, Z.; Shi, Z. Random Access Memories: A New Paradigm for Target Detection in High Resolution Aerial Remote Sensing Images. *IEEE Trans. Image Process.* **2018**, *27*, 1100–1111. [CrossRef] [PubMed]
26. Xia, G.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A Large-Scale Dataset for Object Detection in Aerial Images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3974–3983. [CrossRef]
27. 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]
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
29. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 142–158. [CrossRef] [PubMed]
30. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [CrossRef]
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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
32. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37.
33. Cai, B.; Jiang, Z.; Zhang, H.; Yao, Y.; Nie, S. Online Exemplar-Based Fully Convolutional Network for Aircraft Detection in Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1095–1099. [CrossRef]
34. Long, Y.; Gong, Y.; Xiao, Z.; Liu, Q. Accurate Object Localization in Remote Sensing Images Based on Convolutional Neural Networks. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 2486–2498. [CrossRef]
35. Li, Q.; Mou, L.; Xu, Q.; Zhang, Y.; Zhu, X.X. R-Net: A Deep Network for Multioriented Vehicle Detection in Aerial Images and Videos. *IEEE Trans. Geosci. Remote Sens.* **2019**, 1–15. [CrossRef]

36. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In *Computer Vision–ECCV 2014*; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 818–833.
37. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
38. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [CrossRef]
39. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141. [CrossRef]
40. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving Into High Quality Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162. [CrossRef]
41. Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv* **2013**, arXiv:1312.4400.
42. Yao, Y.; Jiang, Z.; Zhang, H. High-resolution optical satellite image simulation of ship target in large sea scenes. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 1241–1244. [CrossRef]
43. Chen, T.; Li, M.; Li, Y.; Lin, M.; Wang, N.; Wang, M.; Xiao, T.; Xu, B.; Zhang, C.; Zhang, Z. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *arXiv* **2015**, arXiv:1512.01274.
44. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
45. 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.