BIG MAP R-CNN FOR OBJECT DETECTION IN LARGE-SCALE REMOTE SENSING IMAGES

LINFEI WANG\textsuperscript{a}, DAPENG TAO\textsuperscript{a,}\textsuperscript{*}, RUONAN WANG\textsuperscript{b}
RUXIN WANG\textsuperscript{b}, AND HAO LI\textsuperscript{c}

\textsuperscript{a}FIST LAB, School of Information Science and Engineering, Yunnan University Kunming
\textsuperscript{b}Yunnan Union Vision Technology Co Ltd. Kunming
\textsuperscript{c}School of Software, Yunnan University Kunming, Yunnan University Kunming

(Communicated by Yunlong Feng)

ABSTRACT. Detecting sparse and multi-sized objects in very high resolution (VHR) remote sensing images remains a significant challenge in satellite imagery applications and analytics. Difficulties include broad geographical scene distributions and high pixel counts in each image: a large-scale satellite image contains tens to hundreds of millions of pixels and dozens of complex backgrounds. Furthermore, the scale of the same category object can vary widely (e.g., ships can measure from several to thousands of pixels). To address these issues, here we propose the Big Map R-CNN method to improve object detection in VHR satellite imagery. Big Map R-CNN introduces mean shift clustering for quadric detecting based on the existing Mask R-CNN architecture. Big Map R-CNN considers four main aspects: 1) big map cropping to generate small size sub-images; 2) detecting these sub-images using the typical Mask R-CNN network; 3) screening out fragmented low-confidence targets and collecting uncertain image regions by clustering; 4) quadric detecting to generate prediction boxes. We also introduce a new large-scale and VHR remote sensing imagery dataset containing two categories (RSI LS-VHR-2) for detection performance verification. Comprehensive evaluations on RSI LS-VHR-2 dataset demonstrate the effectiveness of the proposed Big Map R-CNN algorithm for object detection in large-scale remote sensing images.

1. Introduction. Object detection in remote sensing images becomes increasingly important due to its use in a wide range of practical applications such as urban planning [25], [19], ecological monitoring [36], and traffic monitoring [16]. Object detection usually aims to recognize natural or man-made objects in satellite images, a relatively straightforward task when images contain simple scenes with obvious foreground targets. However, remote sensing images tend to have more varied categories, complicated backgrounds, variable objects, and rich features with increasing resolution. As a result, comprehensive remote sensing image analysis remains a significant challenge, especially object detection in remote sensing images, due to difficulties in establishing effective feature extraction and robust detection.

\textsuperscript{*}Corresponding author: Dapeng Tao.

2010 Mathematics Subject Classification. Primary: 68T10, 68T45.
Key words and phrases. Object detection, large-scale remote sensing image, Big Map R-CNN, clustering, VHR.
There have been significant efforts over the last few years to improve the performance of object detection in remote sensing images. Existing methods can be divided into two main categories according to the features used: handcrafted methods and deep learning methods.

The handcrafted methods mainly design various features according to the characteristics of human vision such as color, shape, illumination, and spatial and texture information. Among the commonly used handcrafted methods, color histograms [11] are generally considered to be the most computationally efficient and suitable for tackling problems caused by illumination changes. Scale-invariant feature transform (SIFT) [23] describes local images based on gradient information, and SIFT extensions exist such as PCA-SIFT [18] and SURF [2]. However, SIFT-based methods are highly invariant to image changes regardless of scale and rotation. Texture descriptors [26] utilize features such as Gabor [20] and local binary patterns (LBP) [30] to represent relative differences between local satellite images. GIST descriptors [1] provide significant information by extracting local spatial features with a number of pyramid filters and, owing to its computational efficiency, GIST descriptors are widely used in remote sensing image detection tasks. The histogram of oriented gradients (HOG) descriptor [8] is another handcrafted method that captures object edge or local shape features, and HOG has been applied to serial image analysis. Due to the variety of backgrounds usually found in large-scale remote sensing images, a single feature cannot fully describe the entire image. Therefore, the strategy of combining multiple features has been proposed to solve this problem; for example, Ma et al. [24] proposed a locally linear transforming scheme to fuse rigid and nonrigid features for remote sensing image registration. Even though combining multiple features can improve performance, no existing fusion model fully balances various types of feature data. Especially in practical applications, handcrafted descriptors are generally inadequate to describe the rich semantic information contained in satellite images.

In order to overcome the shortcomings of handcrafted methods, strategies have been developed to learn high-level feature representations automatically from images. Deep feature learning-based neural network schemes [29] are playing an increasingly prominent role in remote sensing image analysis. Typical deep learning models such as deep belief networks (DBNs) [27] and convolutional neural networks (CNNs) [5] are composed of various processing layers that allow the discovery of more complicated structures and discriminative information. The obtained information has been exploited to represent semantic-level image properties, and recent research on remote sensing images has demonstrated that learning powerful features can help to improve detection performance. Vakalopoulou et al. [35] utilized deep convolutional networks on satellite images for feature extraction, which increased detection accuracy. Zhu et al. [39] designed a new deep convolutional network to achieve orientation-robust detection in aerial images. Li et al. [21] reported sparse corner voting and Cauchy graph optimization for superpixel-based satellite image detection.

As the most successful deep learning scheme, CNNs have performed well in many remote sensing image detection tasks. Cheng et al. [6] proposed a rotation-invariant CNN model to improve the performance of VHR remote sensing image detection. Han et al. [12] proposed a pre-training mechanism to improve the efficiency of multiple features for geospatial object detection in high spatial resolution satellite imagery. Zhong et al. [38] utilized residual networks and a pre-training mechanism
Figure 1. Motivation for the proposed method. (a) Remote sensing scene of Madrid Airport. (b) Remote sensing scene of the South China Sea. These examples are from the RSI LS-VHR-2 dataset. The targets in the images are indicated by red circles. The remote sensing scenes show the characteristics of large scale, high resolution, and relatively sparse target distribution, which means that existing methods are suboptimal for detection.

for position-sensitive balancing. Empirically [3], region-based CNN (R-CNN) [9] methods have been shown to have greater detection accuracy in large-scale satellite images. For example, Kang et al. [17] proposed a contextual R-CNN to fuse lower-level features for ship detection. Yan et al. [37] proposed Cascade R-CNN and modified the IoU-based weighted loss in remote sensing imagery for object detection. Ren et al. [32] modified faster R-CNN to explore feature pyramids in optical remote sensing images. Despite these efforts, the following limitations remain:

1) Many real-life satellite scenes cover a large scale with relatively sparsely distributed targets (Fig. 1). To our best knowledge, existing algorithms focus on large-scale satellite scenes containing large numbers of small targets in crowded neighboring environments.

2) Traditional human-defined regions obtained by the sliding window strategy [3] may not be optimal for remote sensing object recognition, because cropping large-scale remote sensing images into small images to acquire all target-containing proposals, it makes the detection process is time-consuming.

To overcome these problems, here we propose Big Map R-CNN for sparse object detection in large-sale remote sensing images. The system scheme for Big Map R-CNN is shown in Fig. 2. Our proposed method involves three main steps:

1) Small sub-images are generated by large-scale cropping before being sent to the deep learning detection model. The generated sub-images ensure that images are not scaled during training and testing.

2) High-quality region proposals are generated by the Big Map R-CNN, which simultaneously predicts object bounding boxes and scores. In addition, a detection confidence score threshold is set to obtain candidate object areas.

3) Mean shift clustering is then utilized to process the generated candidate areas and, when possible target positions are located, quadric detecting is introduced to rescan areas of interest.

Our contributions can be summarized as follows: 1) Big Map R-CNN is proposed to address sparse object detection in large-scale remote sensing scenes. 2) Batch detection with non-overlapping cropping is helpful for reducing computational
Figure 2. The scheme of Big Map R-CNN, containing three main components: 1) cropping the input big map in the form of a sliding window; 2) detecting each sub-image sequentially and filtering possible object areas; 3) using mean shift clustering to precisely locate candidate object areas, cropping the new sub-images containing possible objects, and using quadric-detecting to judge whether there is an object or not.

complexity. The mean shift clustering method is proposed to locate possible objects areas, and quadric detecting is introduced to ensure that objects are correctly identified. 3) Our comprehensive experiments on our new RSI LS-VHR-2 dataset demonstrate that Big Map R-CNN achieves superior performance to state-of-the-art approaches in large-scale remote sensing images.

The remainder of the paper is organized as follows. In Section II, we briefly present related works on deep learning-based object detection. We detail Big Map R-CNN based on mean shift clustering in Section III. Section IV presents the results of Big Map R-CNN applied to the RSI LS-VHR-2 dataset. Finally, we present our conclusions in Section V.

2. Related work. We first briefly review existing deep learning-based object detection algorithms. These algorithms have been widely used for remote sensing and other practical applications. According to the network design, we simply group these algorithms into two categories: end-to-end learning algorithms and region proposal learning algorithms, as follows.

2.1. End-to-end learning algorithms. Representative end-to-end learning algorithms include You Only Look Once (YOLO) [31] and Single Shot Multi-Box Detector (SSD) [22]. YOLO frames object detection as a regression problem, predicting the bounding boxes and class probabilities from input images in one evaluation using a single neural network. The YOLO model is simple to construct and can be trained directly on full images. However, the loss function in YOLO is insensitive to different sized bounding boxes, which can result in incorrect object localization. Simony et al. [33] used complex YOLO to estimate the pose of 3D objects, which estimated various classes and achieved high accuracy. SSD is faster than YOLO because it eliminates the proposal generation and pixel or feature resampling stages.
In addition, SSD successfully increased the detection accuracy by producing different predictions from different scales of feature maps, and the predictions were separated by the aspect ratio. Jeong et al. [15] proposed an SSD enhancement that changed the structure to effectively utilize feature maps and thereby improve performance.

2.2. Region proposal learning algorithms. Region proposal learning algorithms play a central role in computer vision systems in which the object detection problem is considered using the recognition using regions paradigm. In their pioneering work in 2014, Girshick et al. [9] exploited the R-CNN framework to use a few thousand category-independent region proposals to reduce the search space for an image. However, repeatedly applying deep convolutional networks to thousands of region proposals for each image decreased the efficiency of detection. SPP-net [13] was proposed to resolve the limitation of R-CNN requiring a fixed-size input image by using a spatial pyramid pooling strategy, which generated a fixed-length representation from various input image scales. However, feature computation in SPP-net was time-consuming. Compared with R-CNN and SPP-net, Fast R-CNN [10] significantly improved both detection efficiency and accuracy by pooling and reusing the convolutional layers. Furthermore, the extraction of high-level features on proposal windows was accelerated by using the region of interest (ROI) pooling policy. However, Fast R-CNN fails to detect small objects, when there are considerable overlaps between neighboring regions. Faster R-CNN [33] was subsequently proposed and combined object proposal and detection into a region proposal network (RPN), thereby reducing the number of region proposals to 300. Notably, the smaller number of proposals not only reduced computation time but also achieved higher detection accuracy. R-FCN [7] was also based on region proposal learning and utilized position-sensitive score maps to balance the translation invariance between image classification and object detection. Compared to Faster R-CNN, this method was much faster at detection but at the cost of reduced accuracy. Mask R-CNN [14] was deemed an intuitive extension of Faster R-CNN and added a branch between network inputs and outputs for instance-level semantic segmentation, thereby achieving pixel-to-pixel alignment.

In conclusion, although the above methods were successful for general object detection in natural images, there still exists room for improvement in detection efficiency in large-scale remote sensing images. Inspired by Mask R-CNN, here we introduce Big Map R-CNN to increase the detection speed and the location precision of the objects.

3. Proposed work. In this section, we present our new deep learning scheme, Big Map R-CNN, for object detection in large-scale remote sensing images. In Big Map R-CNN, object detection is performed according to the following three stages: (1) cropping large-scale remote sensing images into several sub-images by utilizing the non-overlapping sliding window mechanism; (2) detecting each sub-image in the Mask R-CNN network sequentially and filtering possible object areas; and (3) using the mean shift cluster algorithm to precisely locate the position of possible objects, generate new sub-images containing the possible objects, and using quadric detecting to judge whether or not they contain objects. Compared with the existing remote sensing image detection methods, Big Map R-CNN has the following characteristics: (1) The detailed features of remote sensing images are retained by batch detection; (2) Cropping original image with non-overlapping sliding window to
reduce the total number of input images, the computational burden of the detector was correspondingly reduced; (3) The clustering-based quadric detection method can re-scan the suspected region to improve the total detection performance.

3.1. Large-scale cropping. Large-scale cropping focuses on preserving more useful information in remote sensing images by utilizing the non-overlapping sliding window method to crop remote sensing images into several small sub-images; this ensures that the remote sensing images are not scaled during training and testing. Specifically, the sliding window size is $N \times N$ pixels and the sliding step is $N$ pixels. Subsequently, we define the generated sub-image as $Q_{ij}$, where $i$ indicates the row index in the initial image and $j$ is the column index in the initial image. We then send all sub-images to the deep learning network for detection.

3.2. Sub-image detection. Cropping large-scale images directly may divide objects into different sub-images, which will decrease detection accuracy. To fully capture objects, we perform sub-image detection as follows. 1) First, we send all sub-images into the CNN for feature extraction, the generated feature maps providing useful feature information of remote sensing images. 2) In the RPN net, we set the feature maps as input and then output different region proposal sizes. Notably, bounding box regression is introduced to obtain as accurate region proposals as possible. 3) We combine the region proposals with the last layer feature map to the RoI pooling layer, acquiring the feature map of the target area for subsequent classification and positioning. 4) We then classify the objects contained in the target area and filter possible objects. Specifically, Algorithm 1 is the pseudocode for possible object selection.

3.3. Quadric detecting. To locate possible objects accurately, we introduce a clustering method to process candidate sub-images. Considering there exists a possible object in $Q_{ij}$, we define $r_i$ as the area containing all region proposals of sub-image $Q_{ij}$ and its neighboring sub-images. To find the centroid position $q_{\text{center}}(x_i, y_j)$ of the possible object, a baseline centroid position of the region proposals is randomly selected, denoted as $q$. The kernel density estimation function about $q$ can be computed by:

$$
\hat{f}(q) = \frac{1}{mrv^2} \sum_{t=1}^{m} K({q - \hat{q}_{ij}}/r),
$$

(1)
**Algorithm 1** Pseudocode for selecting the possible object

| Input: | large-scale remote sensing image |
|---|---|
| **Parameter:** |  |
| \( Q_{ij} \): sub-image of row \( i \), column \( j \). |  |
| \( \tau \): the threshold to evaluate whether the generated sub-images contain an object or not. |  |
| \( P_{ij} \): the probability score of objects contained in sub-image \( Q_{ij} \). |  |
| \( L \): the detection bounding box position of possible object, \( L = (x, y, w, h) \), where \( x \) is the X-axis coordinate of center point of \( L \) and \( y \) is the Y-axis coordinate, \( w \) is the width of \( L \), \( h \) is the height of \( L \). |  |
| **Output:** | the candidate areas of possible objects. |

**Step 1:** Input sub-images for feature extraction.

**Step 2:** Utilize RPN net to generate region proposals of objects.

**Step 3:** Map the target area to the feature map by using the ROI pooling layer.

**Step 4:**

- for each sub-image
  - if \( P_{ij} < \tau \) and the bounding box has one or more side coincidences with the edge of sub-image.
  - Select the neighborhood of sub-images based on the coincident edge.
  - Map sub-image \( Q_{ij} \) and its neighborhood sub-images back to the original input image and then form a new candidate area.
- else judge the existence of complete object.

**Step 5:** Return \( L \)

**Step 6:** end while

where \( m \) is the number of points belong to the circle region centered on \( q \), and \( r \) is the radius. The index \( i \) and \( j \) indicate the sub-image of row \( i \) and column \( j \), respectively. \( K(\cdot) \) is a radially symmetric function. According to the character of kernel function \( k(||q||^2) \), the above kernel density estimation function is calculated as follows:

\[
\hat{f}(q) = \frac{C_k}{mr^2} \sum_{t=1}^{m} K\left(\frac{q - q_{ij}^t}{r} \right)^2, \tag{2}
\]

where \( C_k \) is a normalized constant. Through the kernel density estimation function, we find that each point in the circle region is applied with different influence weight. Notably, the weight factor is measured by the distance from center point \( q \).

\[
\nabla \hat{f}(q) = \frac{c_k}{mr^2} \sum_{t=1}^{m} \nabla K\left(\frac{q - q_{ij}^t}{r} \right) \tag{3}
\]

\[
= 2c_k \frac{mr^4}{mr^4} \sum_{t=1}^{m} (q - q_{ij}^t) K’\left(\frac{q - q_{ij}^t}{r} \right)^2
\]

\[
= 2c_k \frac{mr^4}{mr^4} \left[ \sum_{t=1}^{m} G \left(\frac{q - q_{ij}^t}{r} \right)^2 \right] \left[ \sum_{t=1}^{m} q_{ij}^t G \left(\frac{q - q_{ij}^t}{r} \right)^2 - q \right].
\]

We next need to find the distribution of the maximum density data based on the probability density estimation function. By taking the derivative of the density function and simultaneously paying attention to the gradient changes, the process is formulated as (3), where the Gaussian kernel function \( G(q) = -K'(q) \). We express
Table I. DESCRIPTION OF THE RSI LS-VHR-2 DATASET

| Label Name | Total instances | Complete instances | Fragmentary instances | Scene class Images | Image width | Sub-images |
|------------|-----------------|--------------------|-----------------------|-------------------|-------------|------------|
| aircraft   | 103917          | 85975              | 17942                 | 203               | 2858        | 6000-15000 |
| ship       | 68436           | 54386              | 14050                 | 30                | 397         | 5000-18000 |

Letting $\nabla f(q) = 0$, we can obtain the point with the highest probability density, which means that the two-dimensional centroid position of the possible object is:

$$q = \frac{\sum_{t=1}^{m} q_{ij}^{t} G \| q - q_{ij} \|^{2}}{\sum_{t=1}^{m} G \| q - q_{ij} \|^{2}}.$$  

(5)

After locating the two-dimensional centroid position of $r_{i}$, we can crop possible objects from original remote sensing images. The four vertices of the crop box are defined as top left $l_{tl}^i$, bottom left $l_{bl}^i$, top right $l_{tr}^i$, and bottom right $l_{br}^i$, which are computed by:

$$l_{tl}^i = \left( x_i - \frac{N}{2}, y_i + \frac{N}{2} \right),$$

(6)

$$l_{bl}^i = \left( x_i - \frac{N}{2}, y_i - \frac{N}{2} \right),$$

(7)

$$l_{tr}^i = \left( x_i + \frac{N}{2}, y_i + \frac{N}{2} \right),$$

(8)

$$l_{br}^i = \left( x_i + \frac{N}{2}, y_i - \frac{N}{2} \right).$$

(9)

The possible objects located in sub-images are sent to the deep neural network model after mean shift clustering, which enables us to obtain information about whether or not there is an object.

4. Experiments. We conducted extensive empirical evaluations on the RSI LS-VHR-2 dataset to demonstrate the effectiveness of the proposed method by comparing with representative baselines. The evaluation criteria were average precision (AP) and the precision-recall curve (PRC) for each class. In addition, mean AP (mAP) for all classes was computed to evaluate the overall performance of the proposed methods.

4.1. Dataset description. To advance performance evaluation research in remote sensing object detection, we built the Remote Sensing Imagery of Large-Scale-VHR-2 categories (RSI LS-VHR-2) dataset, which is much larger than most existing datasets in this field. Table I lists the details of the dataset for two categories, aircraft and ship. As shown in Table I, the RSI LS-VHR-2 dataset has four notable characteristics:
Figure 4. Some examples from the RSI LS-VHR-2 dataset.

1) **Rich image variability**: this dataset is collected from different sensors and platforms and includes 203 airports and 30 harbors.

2) **Large scale**: the width and height of each original image varies from 5000 to 18,000 pixels and contains objects exhibiting a wide variety of scales, orientations, and shapes.

3) **Abundant instances**: the dataset consists of 172,353 positive samples (103,917 aircrafts and 68,436 ships) obtained from 3255 large-scale remote sensing images distributed in 115,989 sub-images cropped from the original large-scale images.

4) **Multiple target difference**: an additional 31,992 fragmented instances were added to the dataset for data augmentation to test the capacity of trained models to detect incomplete targets.

All the original large-scale images were cropped with a non-overlapping sliding window to generate sub-images. To facilitate feature extraction, the sub-image size is a uniform $600 \times 600$ pixels.
Table II. DETAILS OF THE TEST IMAGES

| Label | Scale(pixels) | Images Instances | Sub-images |
|-------|---------------|------------------|------------|
| aircraft | 8000 × 8000 | 5 | 272 | 980 |
| ship | 8000 × 8000 | 5 | 225 | 980 |

4.2. Implementation details. In order to verify the effectiveness of our method, we compared Big Map R-CNN with typical one-stage and two-stage object detection methods including Mask R-CNN [14], Faster R-CNN [10], and YOLOv3 [31]. The body network architecture of YOLOv3 was darknet-53, while all the other experiments were based on the widely used network model ResNet50 [3], which were pre-trained on RSI LS-VHR-2 dataset for image feature extraction. For fair comparison, we selected 10 large-scale VHR remote sensing images as test images, and each large-scale VHR image could be divided into 196 600-pixel sub-images. The details of the large-scale images are given in Table II, and we utilized bounding boxes to annotate the objects as the ground truth. After extracting image features, Big Map R-CNN and the typical object detection algorithms based on CNNs were trained to detect aircrafts and ships by computing feature maps.

To further improve generalization performance, the detection network was fine-tuned for 12,000 iterations. The initial learning rate was 0.0025, and Stochastic Gradient Descent (SGD) was used in the network. The weight decay and momentum were set to 0.0001 and 0.9, respectively, in the iteration process. All experiments were implemented on a computer with 2 Intel Xeon E5-2620 2.1GHz, 8-core CPUs, 8 NVIDIA GTX1080TI GPUs, and 128 GB memory.

Before the comparisons between proposed method and other methods, we set proper parameters for several methods. The image size of region proposal network-based methods (i.e., Mask R-CNN and Faster R-CNN) is set to 600*600, and YOLOv3 is set to 416*416. The images number per batch of all methods is set to 8 due to the limitation of GPU memory. Moreover, we set random to 1 for improve the robustness of the model in YOLOv3. The other parameters are consistent with default setting in original papers [14], [31], [33]. All the region proposal network-based methods are implemented with PyTorch 1.0 and YOLOv3 is implemented with darknet.

4.3. Evaluation metrics. Two commonly used evaluation metrics, precision recall curve (PRC) and mean average precision (mAP), were adopted to quantitatively analyze the detection results of the four methods.

**PRC:** precision recall curve is associated with the number of true positives (TP), the number of false positives (FP), and the number of false negatives (FN). The precision and recall metrics are defined as:

\[
\text{Precision} = \frac{TP}{TP + FP},
\]

\[
\text{Recall} = \frac{TP}{TP + FN},
\]

\[
mAP = \frac{\sum_{i}^{n_{cls}} AP_i}{n_{cls}}.
\]

**MAP:** the average precision is defined as the ratio of correctly detectable objects for each class in the images, computes the area under the PRC, and the mAP is obtained by accumulating the mean ratio of the correct objects in the images
regardless of category. Specifically, the mAP is calculated in (12), where \( n_c \) is the number of categories and \( AP_i \) is the average precision of each class. The mAP for each method is presented in Table VI.

4.4. Experimental results and analysis. Firstly, we compared three cropping size of 300*300 (C300), 600*600 (C600) and 800*800 (C800) to analyze the performance of sub-images decomposition in large scale remote sensing images. From Table III, we can observe that C600 achieve much better performance than C300 and C800 in terms of AP values. Moreover, the cost time in RSI LS-VHR-2 test dataset is considered acceptable. As a result, our detection tasks in large-scale remote sensing images were implemented with the cropping size of 600*600.

We then discuss the overall results and then summarize several aspects of these results in detail. Usually, an intersection over union (IoU) threshold is required to identify positive samples and negative samples. Here we conducted comparative experiments with IoU of 0.5 and 0.75 to evaluate the accuracy of bounding box regression. The PRCs of the four detection methods are shown in Fig. 5. (a) and (b) are the PRCs for aircraft, (c) and (d) are the PRCs for ship.

![Figure 5](image_url)

**Figure 5.** PRCs of the proposed Big Map R-CNN method and three other state-of-the-art detection methods (YOLOv3, Faster R-CNN, and Mask R-CNN). (a) is the PRC of the four methods for aircraft when IoU=0.5; (b) is the PRC of the four methods for aircraft when IoU=0.75; (c) is the PRC of the four methods for ships when IoU=0.5; (d) is the PRC of the four methods for ships when IoU=0.75.
Table III. PERFORMANCE COMPARISONS OF THREE DIFFERENT CROPPING SIZE IN Faster R-CNN NETWORK

| Cropping Size | AP  | Cost time(s) |
|---------------|-----|--------------|
| C300          | 0.430 | 45.82       |
| C600          | 0.651 | 13.20       |
| C800          | 0.647 | 8.79        |

Table IV. PERFORMANCE COMPARISONS OF THE FOUR METHODS ON AIRCRAFT

| Method       | IoU=0.5 | IoU=0.75 |
|--------------|---------|----------|
|              | TP      | FP      | FN      | Recall | Precision | AP   | TP      | FP      | FN      | Recall | Precision | AP   |
| YOLOv3       | 213     | 25      | 59      | 0.783  | 0.895     | 0.727 | 166     | 72      | 106     | 0.610  | 0.6974    | 0.494 |
| Faster R-CNN | 242     | 55      | 30      | 0.890  | 0.815     | 0.830 | 189     | 108     | 83      | 0.695  | 0.636     | 0.618 |
| Mask R-CNN   | 245     | 38      | 27      | 0.901  | 0.866     | 0.843 | 184     | 99      | 88      | 0.676  | 0.650     | 0.570 |
| Big Map R-CNN| 261     | 4       | 11      | 0.900  | 0.985     | 0.985 | 241     | 24      | 31      | 0.886  | 0.909     | 0.850 |

Table V. PERFORMANCE COMPARISONS OF THE FOUR METHODS ON SHIP

| Method       | IoU=0.5 | IoU=0.75 |
|--------------|---------|----------|
|              | TP      | FP      | FN      | Recall | Precision | AP   | TP      | FP      | FN      | Recall | Precision | AP   |
| YOLOv3       | 128     | 53      | 97      | 0.569  | 0.707     | 0.513 | 66      | 115     | 59      | 0.293  | 0.365     | 0.213 |
| Faster R-CNN | 164     | 185     | 61      | 0.729  | 0.470     | 0.651 | 78      | 271     | 147     | 0.347  | 0.223     | 0.259 |
| Mask R-CNN   | 166     | 121     | 59      | 0.738  | 0.578     | 0.661 | 78      | 209     | 147     | 0.347  | 0.272     | 0.273 |
| Big Map R-CNN| 191     | 49      | 34      | 0.849  | 0.796     | 0.826 | 133     | 107     | 92      | 0.591  | 0.554     | 0.546 |

Fig. 4 highlights the following trends. In all scenarios, Big Map R-CNN narrowly outperformed Mask R-CNN and Faster R-CNN. Of note, increasing the IoU from 0.5 to 0.75 decreased the performance of all four methods. Our method had a smoother curve than the other three with IoU=0.75. However, the curves of YOLOv3, Faster R-CNN, and Mask R-CNN declined rapidly under the same circumstances.

To further quantify the detection performance of the four methods, six commonly used evaluation metrics were assessed: TP, FP, FN, Recall, Precision, and AP (used in Table IV and V). In general, the average precision for aircraft was higher than for ships in our test images. This was because the target size for ships varied considerably, ranging from 5 pixels to 2000 pixels in the same VHR satellite image. This characteristic requires that the detection methods are highly robust to object size.

In order to illustrate the credibility of proposed Big Map R-CNN, we compared the AP values of Mask R-CNN and proposed method under the same parameter settings. The detailed parameters are list in Table VI and the performance comparisons of two methods are list in Table VII.

We further analyzed the mAP values and inference time of the four detection methods in Table VIII. Its not surprise that Big Map R-CNN greatly outperforms the several mainstream deep learning-based methods in terms of mAP. Especially
when the IoU=0.75, our method improved detection performance by 27.8% compared with Mask R-CNN for large-scale test images, confirming that Big Map R-CNN improves accuracy compared with other deep learning-based methods, which relies solely on bounding box regression to localize objects.

However, we observed that the inference time of Big Map R-CNN achieves the 16 seconds for a large-scale test image of 8000*8000 sized. It could be attribute to the process of quadric detection. In order to improve the detection accuracy of large-scale remote sensing images, Big Map R-CNN performs additional sub-image detection. It is worthwhile to sacrifice a small amount of time to greatly improve the detection performance. Moreover, YOLOv3 has an overwhelming advantage compared with the other three methods in inference times. It confirms that one-stage methods still outperform the two-stage methods in detection speeds. We will also focus on improving the detection efficiency of Big Map R-CNN in the following work.

Finally, we compare the detection results for Mask R-CNN and Big Map R-CNN in Fig. 6. The false alarm rate and missed detection rate in (a) and (c) (Mask R-CNN) were higher than that in (b) and (d) (Big Map R-CNN), especially in the area with dense aircraft and ships. These figures show that Big Map R-CNN has better detection performance compared with Mask R-CNN in the same large-scale remote sensing image.

4.5. Conclusion. In this paper, we identify object fragmentation caused by image cropping as a crucial obstacle to improve the performance of multiscale object detection in large-scale remote sensing images. To address this, we propose the
Big Map R-CNN method, which applies a mean shift clustering algorithm to fragmented objects and boundary objects after the output layer of typical Mask R-CNN. We can locate the areas where the bounding boxes deviate from the ground truth and missed detections by clustering centers and then quadric detecting these image regions. Utilizing a series of quadric-detected low confidence targets produces a highly effective detection network that more accurately detects boundary objects in sub-images, especially under circumstances in which bounding box regression has a higher weight.

Compared to typical detection algorithms based on CNN such as YOLOv3, Faster R-CNN, and Mask R-CNN, Big Map R-CNN shows competitive performance for large-scale remote sensing image detection. In future work, we will focus on improving the detection accuracy of targets with large-scale differences and dense distributions.
Acknowledgments. This research was partially supported by National Natural Science Foundation of China under Grant 61772455 and Grant U1713213, Yunnan Natural Science Funds under Grant 2018FY001(-013) and Grant 2019FA-045, the Program for Excellent Young Talents of National Natural Science Foundation of Yunnan University (2018YDJQ004) and the Major Project of Yunnan Department of Science and Technology (2019CLJY06).

REFERENCES

[1] U. R. Acharya, H. Fujita and S. Bhat, Decision support system for fatty liver disease using GIST descriptors extracted from ultrasound images, *Information Fusion*, (2016), 32–39.

[2] H. Bay, T. Tuytelaars and L. Van Gool, Surf: Speeded up robust features, *European Conference On Computer Vision*, 3951 (2006), 404–417.

[3] Y. S. Cao, X. Niu and Y. Dou, Region-based convolutional neural networks for object detection in very high resolution remote sensing images, 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, (2016), 548–554.

[4] K. Chatfield, K. Simonyan and A. Vedaldi, Return of the devil in the details: Delving deep into convolutional nets, *proceedings of BMVC*, (2014).

[5] L. C. Chen, G. Papandreou and I. Kokkinos, Semantic image segmentation with deep convolutional nets and fully connected crfs, *arXiv:1412.7062*.

[6] G. Cheng, P. Zhou and J. Han, Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images, *IEEE Transactions on Geoscience and Remote Sensing*, 54 (2016), 7405–7415.

[7] J. Dai, Y. Li, K. He and J. Sun, R-fcn: Object detection via region-based fully convolutional networks, *Advances in Neural Information Processing Systems*, (2016), 379–387.

[8] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, *international Conference on Computer Vision & Pattern Recognition*, (2005), 886–893.

[9] R. Girshick, J. Donahue, T. Darrell and J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2014), 580–587.

[10] R. Girshick, Fast R-CNN, *Proceedings of the IEEE International Conference on Computer Vision*, (2015), 1440–1448.

[11] D. Gray and H. Tao, Viewpoint invariant pedestrian recognition with an ensemble of localized features, *Proceedings of the European Conference on Computer Vision*, 5302 (2008), 262–275.

[12] X. Han, Y. Zhong and L. Zhang, An efficient and robust integrated geospatial object detection framework for high spatial resolution remote sensing imagery, *Remote Sensing*, 9 (2017), 666–687.

[13] K. He, X. Zhang, S. Ren and J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, *ECCV*, 8591 (2014), 346–361.

[14] K. He, G. Gkioxari, P. Dollár and R. Girshick, Mask r-cnn, *Proceedings of the IEEE international conference on computer vision*, (2017), 2961–2969.

[15] J. Jeong, H. Park and N. Kwak, Enhancement of SSD by concatenating feature maps for object detection, *BMVC*, (2017), 1–12.

[16] K. Kanistras, G. Martins and M. J. Rutherford, Survey of unmanned aerial vehicles (UAVs) for traffic monitoring, *Handbook of Unmanned Aerial Vehicles*, (2016), 2643–2666.

[17] M. Kang, K. Ji and X. Leng, Contextual region-based convolutional neural network with multilayer fusion for SAR ship detection, *Remote Sensing*, (2017), 860–873.

[18] Y. Ke and R. Sukthankar, PCA-SIFT: A more distinctive representation for local image descriptors, *CVPR*, (2004), 506–513.

[19] S. Khanal, J. Fulton and S. Shearer, An overview of current and potential applications of thermal remote sensing in precision agriculture, *Computers and Electronics in Agriculture*, 139 (2017), 22–32.

[20] V. Kyrki, J. K. Kamarainen and H. Kälviäinen, Simple Gabor feature space for invariant object recognition, *Pattern Recognition Letters*, 25 (2004), 311–318.

[21] Y. Li, Y. Tan and J. Deng, Cauchy graph embedding optimization for built-up areas detection from high-resolution remote sensing images, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8 (2015), 2078–2096.
[22] W. Liu, D. Anguelov and D. Erhan, Ssd: Single shot multibox detector, European Conference on Computer Vision, 9905 (2016), 21–37.

[23] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, 60 (2004), 91–110.

[24] J. Ma, H. Zhou and J. Zhao, Robust feature matching for remote sensing image registration via locally linear transforming, IEEE Transactions on Geoscience and Remote Sensing, 53 (2015), 6469–6481.

[25] M. Mazhar Rathore, A. Ahmad and A. Paul, Urban planning and building smart cities based on the internet of things using big data analytics, Computer Networks, 101 (2016), 63–80.

[26] B. S. Manjunath, J. R. Ohm and V. V. Vasudevan, Color and texture descriptors, IEEE Transactions on Circuits and Systems for Video Technology, 11 (2004), 91–110.

[27] V. Nair and G. E. Hinton, 3D object recognition with deep belief nets, Advances in Neural Information Processing Systems, (2009), 1339–1347.

[28] H. Noh, S. Hong and B. Han, Learning deconvolution network for semantic segmentation, Proceedings of the IEEE International Conference on Computer Vision, (2015), 1520–1528.

[29] W. Ouyang, X. Wang and X. Zeng, Deepd-net: Deformable deep convolutional neural networks for object detection, The IEEE Conference on Computer Vision and Pattern Recognition, (2015), 2403–2412.

[30] M. T. Pham, G. Mercier and O. Regniers, Texture retrieval from VHR optical remote sensed images using the local extrema descriptor with application to vineyard parcel detection, Remote Sensing, 8 (2016), 368–388.

[31] J. Redmon, S. Divvala and R. Girshick, You only look once: Unified, real-time object detection, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2016), 779–788.

[32] Y. Ren, C. Zhu and S. Xiao, Small object detection in optical remote sensing images via modified faster R-CNN, Applied Sciences, 8 (2018), 813–823.

[33] S. Ren, K. He and R. Girshick, Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in Neural Information Processing Systems, (2015), 91–99.

[34] M. Simony, S. Milzy and K. Amendey, Complex-YOLO: An Euler-region-proposal for real-time 3D object detection on point clouds, Proceedings of the European Conference on Computer Vision, 11127 (2018), 197–209.

[35] M. Vakalopoulou, K. Karantzalos and N. Komodakis, Building detection in very high resolution multispectral data with deep learning features, 2015 IEEE International Geoscience and Remote Sensing Symposium, (2015), 1873–1876.

[36] K. S. Willis, Remote sensing change detection for ecological monitoring in United States protected areas, Biological Conservation, 182 (2015), 233–242.

[37] J. Yan, H. Wang and M. Yan, IoU-adaptive deformable R-CNN: Make full use of iou for multi-class object detection in remote sensing imagery, Remote Sensing, (2019), 286–306.

[38] Y. Zhong, X. Han and L. Zhang, Multi-class geospatial object detection based on a position-sensitive balancing framework for high spatial resolution remote sensing imagery, ISPRS Journal of Photogrammetry and Remote Sensing, 138 (2018), 281–294.

[39] H. Zhu, X. Chen and W. Dai, Orientation robust object detection in aerial images using deep convolutional neural network, 2015 IEEE International Conference on Image Processing, (2015), 3735–3739.

E-mail address: 1412518619@qq.com
E-mail address: dapeng.tao@gmail.com
E-mail address: 1871962809@qq.com
E-mail address: rosinwang@gmail.com
E-mail address: lihao707@ynu.edu.cn