FMSSD: Feature-Merged Single-Shot Detection for Multiscale Objects in Large-Scale Remote Sensing Imagery

Peijin Wang, *Student Member, IEEE*, Xian Sun, *Senior Member, IEEE*, Wenhui Diao, *Member, IEEE*, and Kun Fu, *Member, IEEE*

***Abstract*— Recently, the deep convolutional neural network has brought great improvements in object detection. However, the balance between high accuracy and high speed has always been a challenging task in multiclass object detection for large-scale remote sensing imagery. One-stage methods are more widely used because of their high efficiency but are limited by their performances on small object detection. In this article, we propose a unified framework called feature-merged single-shot detection (FMSSD) network, which aggregates the context information both in multiple scales and the same scale feature maps. First, our network leverages the atrous spatial feature pyramid (ASFP) module to fuse the context information in multiscale features by using feature pyramid and multiple atrous rates. Second, we propose a novel area-weighted loss function to pay more attention to small objects, while the replaced original loss treats all objects equally. We believe that small objects should be given more weight than large objects because they lose more information during training. Specifically, a monotonic decreasing function about the area is designed to add weights on the loss function. Extensive experiments on the DOTA data set and NWPU VHR-10 data set demonstrate that our method achieves state-of-the-art detection accuracy with high efficiency. We also build a new large-scale data set called AIR-OBJ data set from Google Earth and show the detection results of small objects, which validates the effectiveness on large-scale remote sensing imagery.**

***Index Terms*— Area-weighted, context information, one-stage, remote sensing imagery, small object detection.**

I. Introduction

With the development of aerospace remote sensing technology, a great many high satellites and images with high spatial or spectral resolution can be obtained easily, and object detection of remote sensing imagery has become one of the hottest topics. Although many methods have been proposed, object detection in large-scale remote sensing images is pretty challenging due to complex scenes and multiscale objects.

Benefitting from the development of deep convolutional neural networks, many methods based on deep learning have been proposed to enhance the accuracy and speed of detection in the field of computer vision [1]. Current deep learning-based detectors can be roughly divided into two categories: two-stage methods and one-stage methods. The current state-of-the-art object detectors are based on two-stage methods, such as R-CNN [2], Fast R-CNN [3], Faster R-CNN [4], and R-FCN [5]. In general, they propose a region proposal network (RPN) to generate high-quality proposals using a deep convolutional neural network in the first stage. Then, the proposals are further used for classification and localization in the second stage. They can achieve relatively good performance, but they are resource-consuming and time-consuming, while one-stage methods have a simple network, such as YOLO [6], SSD [7], and RetinaNet [8], which can predict the localization and classification directly using dense sampling without region proposal module. Compared with two-stage methods, one-stage methods can achieve higher computational efficiency and are more suitable for real-time applications, especially, SSD achieves better unification of accuracy and efficiency. These deep learning-based object detection methods have been widely used in natural images.

Compared with natural images, the scales of remote sensing images are so large that the memory consumption and computation time increase quadratically [9]. Moreover, remote sensing images are more complex because they consist of more multiscale objects in the ground, and the small objects account for a larger proportion in remote sensing imagery. Although the deep learning-based methods have shown good performances on natural images, they do not perform very well on remote sensing images, especially for small objects. The large scale of the input image and the small objects in complex scenes are two considerable obstacles, which have restricted the development of detection methods [10]–[14].

To solve the aforementioned problems, many deep-learning object detection algorithms for remote sensing images have been proposed [15]–[21]. For example, Chen *et al.* [15] propose a vehicle detection model based on sliding windows, which is called a hybrid deep neural network. And

Manuscript received June 21, 2019; revised August 25, 2019; accepted October 12, 2019. This work was supported by the National Science Fund for Distinguished Young Scholars under Grant 61725105. *(Corresponding author: Xian Sun.)*

P. Wang is with the Key Laboratory of Network Information System Technology (NIST), Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100864, China, and also with the School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Science, Beijing 100049, China (e-mail: wangpeijin17@mails.ucas.ac.cn).

X. Sun, W. Diao, and K. Fu are with the Key Laboratory of Network Information System Technology (NIST), Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China (e-mail: sunxian@mail.ie.ac.cn; dwh1031@gmail.com; fukun@mail.ie.ac.cn).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2019.2954328

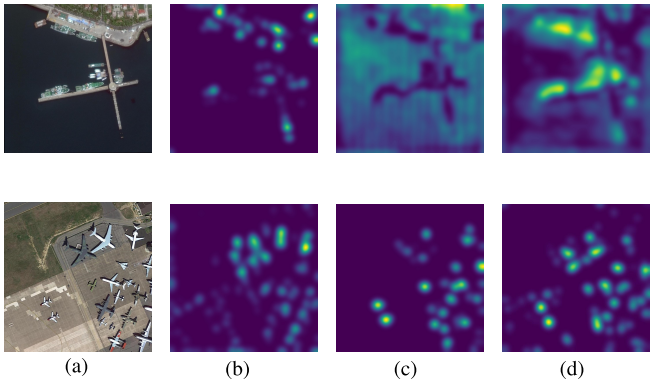


Fig. 1. Visualization of features. The first row shows the impact of the surrounding information of objects. The second row shows the impact of the information of small objects. The merged feature contains more discriminative information, which fuses the original feature with context information. (a) Image. (b) Original feature. (c) Context information. (d) Merged feature.

its contribution is to replicate the convolution layers at different scales and detect the vehicles in different scales. Ammour *et al.* [16] segment the input image into small homogeneous regions, which can be used as candidate locations to train a support vector machine (SVM) classifier. This classifier is used to classify regions into "car" and "no-car" classes. Diao *et al.* [17] use visual saliency to generate a few bounding boxes and then extract features by deep belief networks. This method is only fit for simple scenes. Guo *et al.* [18] propose a multiscale object proposal network and a multiscale object detection network, which shares a multiscale base network. They use the former to generate object proposals and the latter to train a good detector. Han *et al.* [19] combine the robust properties of a transfer mechanism and the sharable properties of Faster R-CNN in the method, which uses a pretraining mechanism to enhance the efficiency of the multiclass object detection. Cheng *et al.* [20] propose a RICNN model by learning a new rotation-invariant layer based on the existing CNN architectures, which is trained by optimizing a new objective function via imposing a regularization constraint. This model deals with the problem of object rotation variation well. Wang *et al.* [21] adopt an end-to-end multiscale visual attention networks method. For each scale feature map, the method learns a visual attention network, which is to highlight the features from object region and suppress the cluttered background.

However, these methods do not perform very well in complex scenes, especially for small object detection in large-scale remote sensing images. Recently, some methods have proved that abundant context information plays an important role in detecting visually impoverished objects, such as small objects or occluded objects. Feature pyramid network (FPN) [22] designs a feature pyramid structure based on two-stage methods, which can combine high-resolution feature maps with low-resolution feature maps. DSSD [23] proposes an encoder-decoder hourglass structure to propagate semantic information before prediction and uses ResNet instead of VGGNet. Recurrent detection with activated semantics (RDAS) [24] uses a segmentation branch to enrich the semantic information at low-level feature maps and several global activation blocks to enrich the semantic information from higher level layers. ICN [25] proposes a novel joint feature pyramid and image cascade network, which extracts information on a wide range of scales and enhances the detection results. Yan *et al.* [26] propose an IoU-Adaptive Deformable R-CNN, which has achieved the best accuracy in all collected articles. They reduce the loss of small object information by a new IoU-guided detection network. Sun *et al.* [27] propose a salience biased loss based on RetinaNet, which uses the salience information of the input image to improve the detection accuracy. The loss treats all input images differently, which focuses on training on a set of complicated images and prevents the vast number of easy cases from guiding the loss of the detector during training. VSSA-NET [28] proposes a multiresolution feature learning module and a vertical spatial sequence attention module to gain more effective features and context information for traffic sign detection. It is distinctive that authors treat the traffic sign detection as a spatial sequence classification and regression task. Scale-transferrable detection network (STDN) [29] designs a scale-transfer module and embeds this module directly into a DenseNet. This method uses the scale-transfer layer in super-resolution to expand the resolution of the feature map for object detection. Wang *et al.* [30] propose a multiview-based parameter-free (MPF) framework to detect coherent groups. They build the orientation and context graphs to perceive the individual's relationship on both the microcosmic and macroscopic views. Deep layer aggregation (DLA) [31] designs DLA architectures which encompass and extend densely connected networks and FPNs with hierarchical and iterative skip connections.

The above-mentioned algorithms consider that the surrounding information of objects can guide the detection of objects, so they extract the context information by fusing features on different scales. As shown in Fig. 1(c), the context information in the first row means the ocean and harbor information. When fusing the surrounding information with the original feature, most ships can be detected accurately. In practical experiments, large objects have more feature information than small objects in the same feature maps, and the features of small objects are likely to vanish as the network going deeper. Hence, it is inadequate to only fuse surrounding information for multiscale objects in remote sensing images. Furthermore, the context information resulting from the difference between multiscale objects cannot be neglected. The second row in Fig. 1 shows that the original feature loses the information of small planes. But if we combine it with the feature of small planes as shown in Fig. 1(c), the detector can detect planes with a whole scaled detection as shown in Fig. 1(d). In this case, the context information can be obtained by giving larger weights on small objects in the same feature maps.

In this article, we propose a feature-merged framework that unifies the context information to improve the detection accuracy. To be specific, we quantify context information in the feature maps and combine it with the popular deep learning-based method SSD [7]. In our network, an atrous spatial feature pyramid (ASFP) module and an area-weighted loss function are designed to enrich the context information. ASFP module consists of several parallel atrous convolution layers with different rates and a feature fusion module. Area-weighted loss, which is a monotone decreasing function

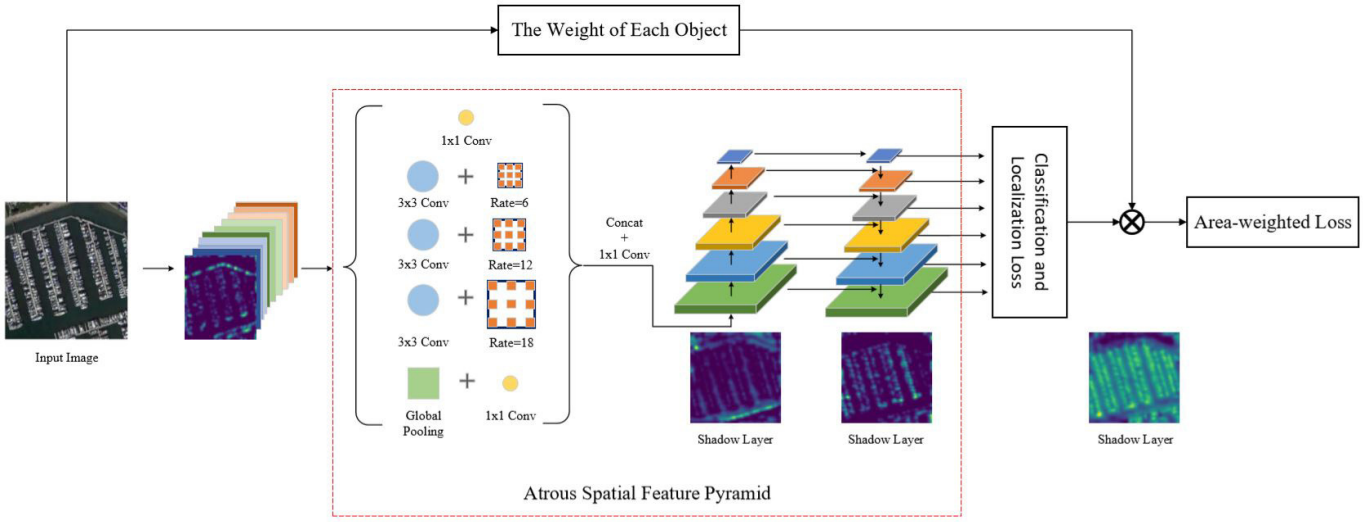


Fig. 2. Pipeline of our FMSSD. It shows the overall architecture of our network, which consists of an ASFP module and an area-weighted loss.

about the size of an object, is used to give larger weight to small objects. The unified framework achieves state-of-the-art performance on two public large-scale data sets. The main contributions of this article are as follows.

1) We present a unified object detection framework for remote sensing images, which can detect multiclass objects in large-scale complex scenes effectively and efficiently. Compared with other detection methods only focusing on context information in multiple scales, our method can quantify the context information in multiscale feature maps as well as different sizes of objects in the same scale feature maps.
2) To obtain the context information in multiple scales, we design an ASFP module which contains several parallel atrous convolution layers with different rates to enlarge the receptive field and a feature fusion module to unify the context information from different feature maps.
3) An area-weighted loss is proposed to guide the network to learn more information about small objects in the same scale feature maps. In our method, each object has a different weight in the loss function, which depends on its area size. Intuitively, the weight increases gradually with the decreasing of the size.

In the horizontal bounding boxes (HBB) tasks of the DOTA data set [32], our model achieves 72.43% mean Average Precision (mAP), and the inference speed is 16 frames/s (FPS). For the NWPU VHR-10 data set, we achieve 90.40% mAP. In the related published articles using these data sets, our method surpasses all one-stage detectors and most two-stage detectors with high inference speed and achieves the best trade-off between accuracy and speed. Furthermore, effectiveness in very large-scale remote sensing images is validated on thousands of images that are 16 393 × 16 393 pixels.

II. Methodology

Remote sensing images contain a lot of small objects whose feature information is often overwhelmed by complex scenes and other large objects. To improve the detection accuracy of small objects, the common way only obtains the context information in multiscale feature maps. We believe that the context information associated with the sizes of objects is also significant for small object detection, which exists in the same scale features. In this article, we propose a unified framework, namely, feature-merged single-shot detection (FMSSD), to extract significant context information. The overall architecture of our network is shown in Fig. 2.

A. Framework

We conclude that the discriminating context information for small object detection is composed of two parts.

1) Context Information Between Multiple Scales: In deep convolutional neural networks, shallow feature maps have rich localization information but less semantic information, and deep feature maps have rich semantic information. For small objects, which are often detected in the shallow feature maps, it is necessary to add semantically strong information surrounding the objects for better detection. Hence, we need to construct the multiscale feature maps with small semantic gaps. A common way is to build a feature pyramid module to extract context information on multiple scales.

2) Context Information in the Same Scale: The size varies dramatically in different categories. For example, a vehicle may be as small as 20 pixels; however, a harbor can be as large as 1000 pixels, which is 50 times larger than a vehicle. All classic object detection methods, like Faster RCNN, SSD, and more recent methods, treat all positives fairly neglecting the size imbalance of target. Collegio *et al.* [34] have demonstrated the impact of object size on visual perception. They find that humans will pay more attention to small size objects in the real world because they need more details for them. Predictions are supported by faster identified targets in objects inferred to be small than large, with costlier attentional shifting in large than small objects when attentional demand was high. The size imbalance causes the problem that training is inefficient as most locations are large-scale objects. Just like in the real

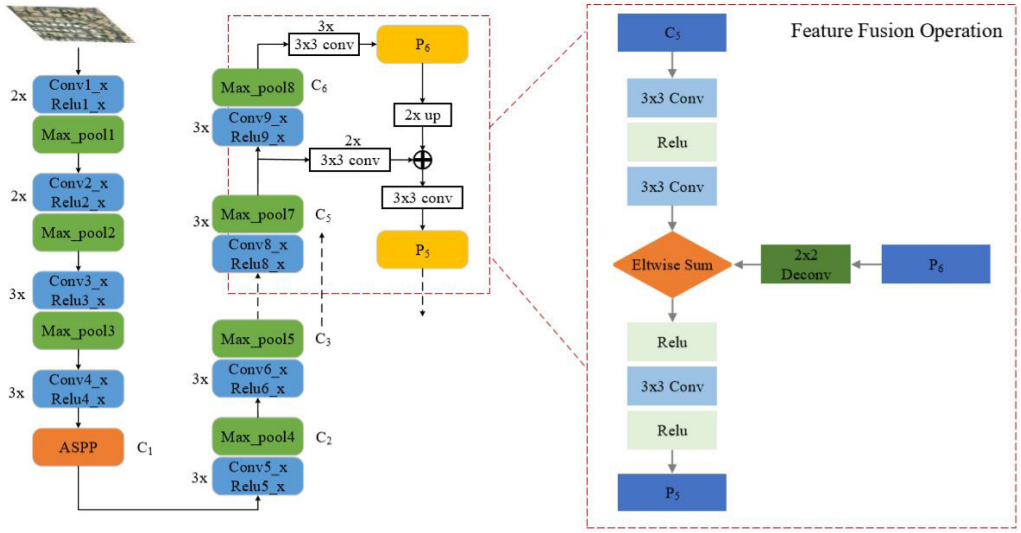


Fig. 3. Detail of ASFP with ASPP module and the feature fusion operation.

Algorithm 1 FMSSD

Input: The bounding boxes of training data B = {B_1, B_2,..., B_n}, the maximum epoch, t, NMS threshold θ, where $0 \leq \theta \leq 1$ and score threshold β, where $0 \leq \beta \leq 1$

Output: Model (detector), and detection results

Step1: *Feature Extractor*
Extract feature by VGG16 [33] model and extra layers
if $out\_stirde = 8$ **then**
  Extract feature by atrous spatial pyramid pooling (ASPP) module
end if
Select six layers to be source features C = {C_1, C_2,..., C_6}
while $i \leq 6$ **do**
  Propagate the feature by feature pyramid module
  Get features pyramid P = {P_1, P_2,..., P_n}
  Predict results R on each feature of P
end while
Step2: *Calculate Loss Function*
Calculate the overlaps over all ground truths B and predict boxes R
if $max\_overlap$>0.5 **then**
  Choose the ground truth to be target and predict box to be positive example
  Calculate the area of target, then calculate the corresponding weight
end if
Calculate the classification loss and localization loss with area weight
Step3: *Process The Detection Results*
Get all detection results
Select the right results by NMS thresh θ and score thresh β

world, we also pay more attention to small objects when observe an image, since they are hard to be distinguished. Hence, we consider that the semantic information in the same feature maps is crucial to small objects. A common solution

TABLE I
ANCHOR SETTING IN EACH LAYERS

Layer name	Output size	Min_size	max_size	Aspect_ratio
P1	64	25.84	76.80	1, 2, 3
P2	32	76.80	153.60	1, 2, 3
P3	16	153.60	230.40	1, 2, 3
P4	8	230.40	307.20	1, 2, 3
P5	4	307.20	384.00	1, 2, 3
P6	2	384.00	460.80	1, 2

is to use augmentation for small objects, which makes a copy of any objects from its original location to different positions [35]. In this way, the contribution of small objects to computing the training loss can be improved, while it is time-consuming and may cover the other objects.

Taking into account the aforementioned analysis, we propose a unified framework for object detection. The proposed approach is built upon the SSD network. Thus, it is considered as a one-stage method. In our network, the feature maps contain more abundant context information and larger receptive fields by using an ASFP module. Moreover, taking into account the context information on the same scale, we design a novel area-weighted loss function that acts as a more effective approach to deal with the size imbalance problem. More details of our method can be seen in Algorithm 1.

B. ASFP Module

It is well known that SSD achieves high computational efficiency, but does not perform very well on small objects, because the shallow layers cannot extract enough context information. SSD uses six feature maps in multiple scales to classify and locate objects [7]. For each feature map, there are many corresponding scales prior to boxes being generated. Generally, small objects are detected in the shallow feature maps, which need abundant context information. As a result, we propose a feature-merged module to propagate the semantic features from a high level to a low level.

TABLE II

NUMBER OF SMALL OBJECTS, TOTAL OBJECTS AND RATIO OF SMALL OBJECTS FOR EACH CLASS

	Plane	BD	Bridge	GFT	SV	LV	Ship	TC
Small objects	10053	339	6782	342	91649	30450	90940	1130
Total objects	37052	3035	9823	2431	98762	60947	122640	12819
Ratio	0.271	0.112	**0.690**	0.141	**0.928**	0.499	**0.742**	0.088
	BC	ST	SBF	RA	Harbor	SP	HC	
Small objects	508	20043	454	1688	5089	4305	3606	
Total objects	3786	27381	2675	3082	20605	6791	4405	
Ratio	0.134	**0.732**	0.170	**0.548**	0.247	**0.634**	**0.819**	

TABLE III

COMPARISON OF THE PERFORMANCE OF EACH DETECTION METHOD ON THE DOTA TESTING DATA SET. EACH EXPERIMENT IS IMPROVED BASED ON THE PRIOR RESULT. THE SHORT NAMES FOR EACH CATEGORY CAN BE FOUND IN SECTION III

Method	Plane	BD	Bridge	GFT	SV	LV	Ship	TC	BC	ST	SBF	RA	Harbor	SP	HC	mAP↑	FPS↓
Baseline	88.23	70.35	34.75	62.24	60.68	62.74	63.65	90.32	75.45	63.89	46.76	58.21	60.13	72.53	49.41	63.96	26
+Focal	87.68	73.09	36.72	63.76	63.89	63.49	65.97	90.73	76.81	64.96	47.89	58.81	62.33	73.78	50.72	64.98	25
+ASPP	88.05	76.75	38.78	62.55	67.21	69.85	71.58	90.67	77.16	69.87	49.91	59.40	63.81	76.14	51.11	67.52	13
+ASFP	88.76	78.64	43.47	61.98	68.35	70.10	74.10	90.67	78.03	68.55	47.10	63.01	69.98	77.65	53.56	68.93	15
+Area_cls	88.78	80.95	46.08	67.71	68.22	72.08	73.49	**90.74**	79.07	71.58	52.43	62.23	71.08	80.68	59.12	71.05	16
+Area_loc	88.61	**81.56**	47.28	67.05	69.96	72.30	75.57	90.71	78.56	72.11	48.91	64.61	70.56	81.36	56.85	71.87	16
+Area_loss	**89.11**	81.51	**48.22**	**67.94**	**69.23**	**73.56**	**76.87**	90.71	**82.67**	**73.33**	**52.65**	**67.52**	**72.37**	**80.57**	**60.15**	**72.43**	16

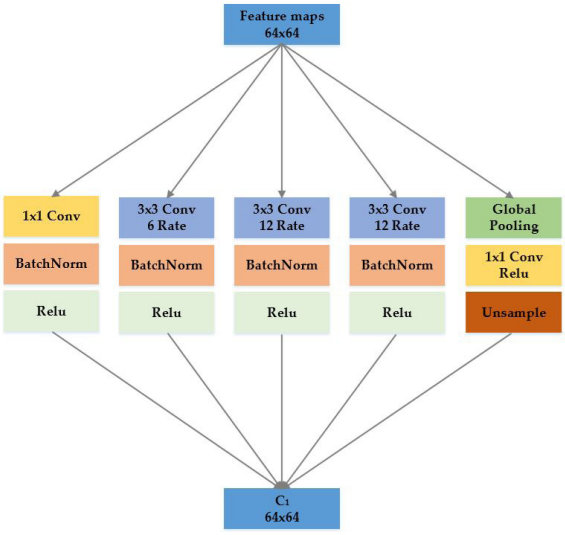


Fig. 4. Detail of the parallel module. This module consists of five branches, which make C1 contain multiscale information.

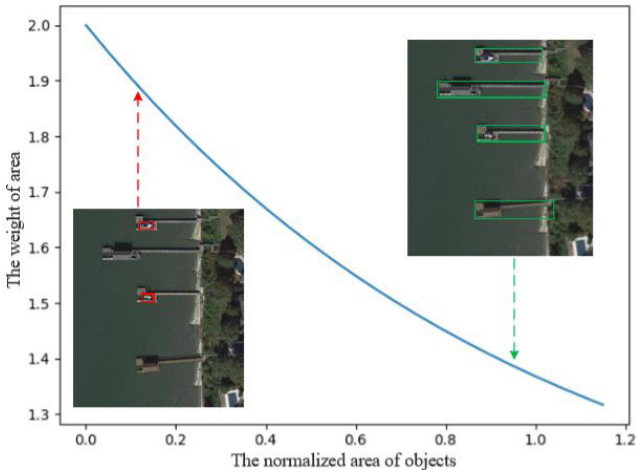


Fig. 5. Graph of area-weight function. Its value is always more than 1, which is decreasing with the increasing of area.

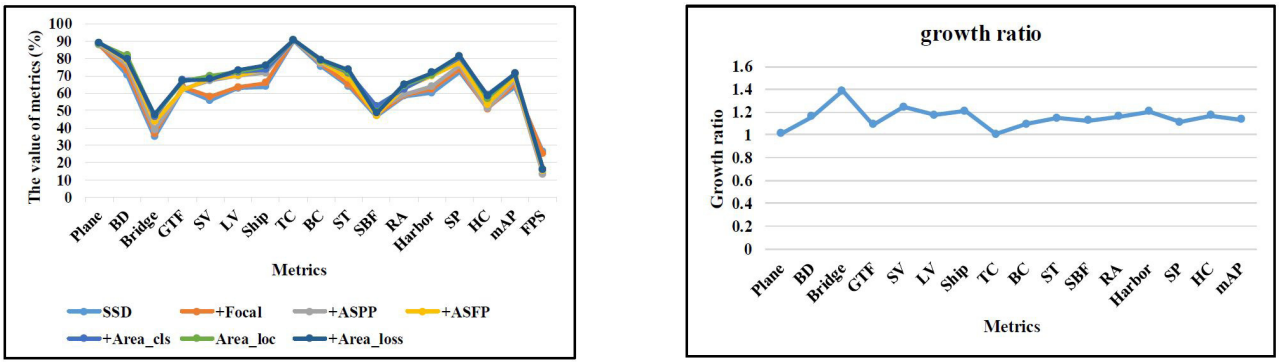
To obtain more informative context information, we use a feature fusion module. After the parallel module, we select six feature maps $\{C_1, C_2, C_3, C_4, C_5, C_6\}$ from the conventional SSD, corresponding to the last layer of each convolutional block, which have strides of {8, 16, 32, 64, 128, 256} pixels with respect to the input image. In the top-down network, we obtain higher resolution feature maps by deconvolutional connections and lateral connections. As shown in Fig. 3, in order to obtain the merged feature P_4, we reduce the channels of C_4 to 256 by two 3×3 convolutional layers, and then we use 2×2 deconvolutional layers for last feature maps P_5. Specially, we use three 3×3 convolution layers for the last feature map C_6 to obtain P_6. Then we merge them by addition, which does not need extra parameters. Specifically, the operations are defined as

$$P_6 = \text{Conv}_{3\times3}(C_6) \tag{1}$$

$$P_i = \text{Conv}_{3\times3}\left(\sum_{j=i+1}^{5} \text{Unsample}(P_j) + \text{Conv}_{3\times3}(C_i)\right). \tag{2}$$

As a result, we can get the final feature maps $\{P_1, P_2, P_3, P_4, P_5, P_6\}$ with rich semantic information. The anchor setting in each layer can be seen in Table I.

The information outside the bounding boxes is significant for classification and localization. Before generating the predicted layers, we use an ASPP module to enlarge the receptive field, which aggregates multiscale context information and global context information to generate more discriminative features. This module consists of a 1×1 convolution, three parallel 3×3 atrous convolutions with different atrous rates



(a) (b)

Fig. 6. Detection results of our method on the DOTA data set. (a) Results of different methods. (b) Growth ratio of each class between our final result and baseline. Figures show the effectiveness of our method, especially on small objects.

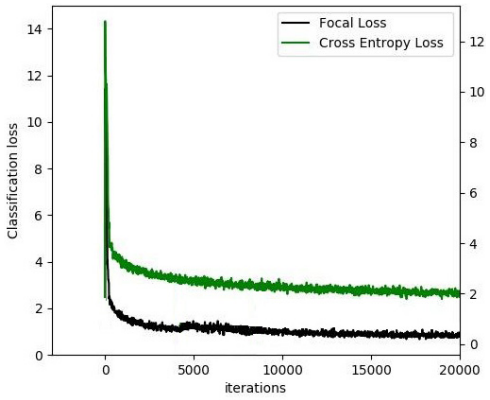


Fig. 7. Variation trend of different classification loss. The classification loss with focal loss decreases rapidly than that without focal loss.

(6, 12, 18) for local context information, and a global pooling layer for global context information, as shown in Fig. 4. The ASPP module is applied on the feature map where the output stride is 8.

We also use a convolutional layer to reduce the number of channels to 256 before prediction, which can decrease the number of model parameters.

C. Area-Weighted Loss Function

In common with other object detection methods, we define an area-weighted loss to combine both classification and localization loss. The area-weighted loss is proposed to improve the object detection scenario which has an extreme imbalance between small objects and large objects in the area (e.g., 1:1000). The loss function in FMSSD is defined as a multitask loss for each image. Area-weighted loss can be described as

$$L_{\text{area-weighted}}(\{p_i\},\{g_i\}) = \frac{1}{N}\left(\sum_i \omega_i L_{\text{cls}}(p_i, p_i^*) + \alpha \sum_i \omega_i t_i L_{loc}(g_i, g_i^*)\right) \quad (3)$$

where i is the index of anchor in one batch, p_i is the predicted probability of anchor i being all kinds of objects, p_i^* is the label of the ground truth bounding box, g_i is a vector representing four parameterized coordinates of predicted bounding box, and g_i^* is that of the ground truth. For training FMSSD, we need to determine the positives and their corresponding ground truths to train the detection network accordingly. We calculate the Jaccard overlap between each default box and ground truth bounding box. We select the ground truth with which the overlap is the largest as its target [7]. As for a default box, if its overlap with target is larger than the threshold (0.5), it is determined as a positive sample. t_i is 1 if the anchor is positive, and 0 if not.

The classification loss L_{cls} is a negative log-likelihood function over all classes, especially, one-stage methods are applied over a dense sampling of object locations, scales, and aspect ratios. There are many easy examples for training because they do not have the second stage to refine the default boxes. If we treat those easy examples similar to the hard examples, those easily classified negative examples would occupy most of the loss and dominate the gradient. We hope that loss could pay more attention to hard examples than easy examples. We use the focal loss to accelerate convergence, which can balance the easy and hard examples [8]. Hence, the classification loss is defined as

$$L_{\text{cls}} = FL(p_t) = -(1-p_t)^{\gamma} log(p_t). \quad (4)$$

We set $\gamma = 2$, which is the same as it is in the task of Pascal VOC.

The localization loss is a smooth L1 loss defined as

$$\text{Smooth}L_1(x) = \begin{cases} 0.5x^2, & \text{if x} < 0 \\ |x| - 0.5, & \text{otherwise} \end{cases} \quad (5)$$

and $t_i * L_{loc}$ means that the localization loss is activated only for positives. The FMSSD regresses to offsets for the center (cx, cy) of the default box (d) and for its height (h) and width (w). This can be thought of as bounding-box regression from an anchor box to a nearby ground truth box

$$\hat{g}_j^{cx} = \frac{\hat{g}_j^{cx} - d_i^{cx}}{d_i^w}, \quad \hat{g}_j^{cy} = \frac{\hat{g}_j^{cy} - d_i^{cy}}{d_i^h}$$
$$\hat{g}_j^{w} = \log\left(\frac{g_j^w}{d_i^w}\right), \quad \hat{g}_j^{h} = \log\left(\frac{g_j^h}{d_i^h}\right). \quad (6)$$

ω_i is the weight of anchor i, if it is set to 1, the network treats all positives fairly neglecting the sizes of targets. In real

world, the attention we pay to multiscale objects is different. A research demonstrates that human brain can adjust various degrees of attention on different objects automatically, which also reflects the detection complexity of objects. It is necessary to pay more attention to small objects while detecting multiscale objects simultaneously.

We hope that ω_i could depend on the size of its target. The size of an object relies on its width and height. However, there are some instances with an extremely large aspect ratio, such as bridges and harbors. When we selected width or height as a weighting factor, it did not work well on these instances. Hence, we use the area of the object to be the weighting factor ω_i for all samples. As for a positive sample, its area-weight depends on the area of its target. Due to the normalization operation, the offset and area are less than 1. Consequently, we design a monotone decreasing function about the area, whose value is always greater than 1. As shown in Fig. 5, we use an exponential function to be our primary function. We apply a larger weight to the predicted box if the area of its target is relatively small. The ω_i is defined as

$$\omega_i = e^{-s_i} + 1. \tag{7}$$

For the ith anchor, s_i is the area of its target. After adding weight ω_i to the loss function, the size of an object has more impact on the loss and gradient. If we put the detail of ω_i into the total loss, the area-weighted loss can be seen as

$$\begin{aligned} &L_{\text{area-weighted}}(\{p_i\}, \{g_i\}) \\ &\quad = \frac{1}{N}\left(\sum_i (e^{-s_i}+1) L_{\text{cls}}\left(p_i, p_i^*\right)\right. \\ &\qquad \left. + \alpha \sum_i (e^{-s_i}+1) t_i L_{\text{loc}}\left(g_i, g_i^*\right)\right). \end{aligned} \tag{8}$$

However, the loss function in conventional object detection networks also consists of classification loss and localization loss, which is defined as

$$L_0(\{p_i\}, \{g_i\}) = \frac{1}{N}\left(\sum_i L_{\text{cls}}\left(p_i, p_i^*\right) + \alpha \sum_i t_i L_{\text{loc}}\left(g_i, g_i^*\right)\right). \tag{9}$$

Compared with the conventional loss, our loss has some extra items

$$\begin{aligned} \Delta L &= L_{\text{area-weighted}}(\{p_i\}, \{g_i\}) - L_0(\{p_i\}, \{g_i\}) \\ &= \frac{1}{N}\left(\sum_i e^{-s_i} L_{\text{cls}}\left(p_i, p_i^*\right) + \alpha \sum_i e^{-s_i} t_i L_{\text{loc}}\left(g_i, g_i^*\right)\right). \end{aligned} \tag{10}$$

$\triangle L$ is related to the size of the target, and small objects have a greater impact on the total loss. Similarly, the size of an object affects the gradient when training, and further affects the prediction of bounding boxes.

III. Experiments

This section provides a concise description of the experimental details, such as data sets, evaluation metrics, implementations details, and experimental results over the public remote sensing object detection data sets.

TABLE IV
APs, APm, and APl of Different Methods on the DOTA Validation Set

Method	APs	APm	Apl
Baseline	48.6	77.3	66.5
+Focal	52.6	76.3	68.9
+ASPP	55.5	76.8	70.4
+ASFP	56.4	77.6	71.2
+Area_loss	**58.2**	**77.9**	**70.6**

TABLE V
Performances of FMSSD Starting From Different Layers

Start layer	con5_3	con6_2	con7_2	con8_2	con9_2
mAP	71.92	71.90	70.77	71.86	**72.43**

A. Data Sets

To advance object detection research in Earth Vision, many public data sets have been published for researchers to conduct further investigations. Some data sets like VEDAI [36], COWC [37], and DLR 3K Munich Vehicle [38] have only the class of vehicles. UCAS-AOD [39] includes vehicles and planes but HRSC2016 [40] only contains ships. All the above-mentioned data sets are not applied in complicated scenes due to the lack of classes. To evaluate the effectiveness of our method, we use three data sets to product experiments.

1) DOTA Data Set: It is a large-scale data set for object detection, which contains 2806 images from different sensors and platforms with crowdsourcing. Each image is sized about $4k \times 4k$ pixels and contains objects of different scales, orientations, and shapes. The fully annotated DOTA data set contains 188 282 instances, annotated by experts in remote sensing image interpretation, with respect to 15 common object categories: plane, baseball diamond (BD), bridge, ground field track (GFT), small vehicle (SV), large vehicle (LV), tennis court (TC), basketball court (BC), storage tank (ST), soccer ball field (SBF), roundabout (RA), swimming pool (SP), and harbor and helicopter (HC). Each of the images is labeled by an arbitrary quadrilateral. The data set randomly selects half of the original images as the training set, 1/6 as the validation set, and 1/3 as the testing set [32].

In this article, we only conduct experiments on HBB and, hence, we select the external rectangle of the ground truth boxes to be the ground truth. We crop images into 1024 $\times$ 1024 pixels with overlap 256 pixels. Finally, we extract 27 406 images for training and 4915 images for evaluating. Following the definition in the DOTA data set [32], we refer to the height of a horizontal bounding box, which we call pixel size for short, and then divide all the objects into three splits according to their height: small for range from 10 to 50 pixels, middle for range from 50 to 300 pixels, and large for range above 300 pixels. Therefore, we define the object whose height ranges from 10 to 50 pixels as a small object. Table II shows the ratios of small objects for different classes in the training set and validation set. We highlight in bold the value of the small-object ratio which is relatively larger than

TABLE VI
COMPARISON OF THE PERFORMANCE OF ONE-STAGE METHODS ON THE DOTA TESTING DATA SET

Method	Plane	BD	Bridge	GFT	SV	LV	Ship	TC	BC	ST	SBF	RA	Harbor	SP	HC	mAP↑	FPS↑	Std↓
DSSD [23]	**91.10**	71.80	**54.60**	66.40	79.00	77.20	87.50	87.60	52.10	69.70	38.00	**72.60**	**75.40**	59.40	28.90	67.40	9	17.38
YOLOv2 [44]	76.90	33.87	22.73	34.88	38.73	32.02	52.37	61.65	48.54	33.91	29.27	36.83	36.44	38.26	11.61	39.20	30	15.18
YOLOv2* [45]	81.30	52.50	36.30	40.00	76.40	76.20	85.60	76.90	45.20	70.70	38.50	63.40	43.90	57.00	42.30	59.10	15	16.97
YOLOv3 [46]	79.00	77.1	33.90	68.10	52.80	52.20	49.80	89.90	74.80	59.20	**55.50**	49.00	61.50	55.90	41.70	60.00	13	14.66
DYOLO [45]	86.60	71.40	54.60	52.50	**79.20**	**80.60**	**87.80**	82.20	54.10	**75.00**	51.00	69.20	66.40	59.20	51.30	68.10	17	13.03
RetinaNet [8]	78.22	53.41	26.38	42.27	63.64	52.63	73.19	87.17	44.64	57.99	18.03	51.00	43.39	56.56	7.44	50.39	14	20.89
SBL [27]	89.15	66.04	46.79	52.56	73.06	66.13	78.66	**90.85**	67.40	72.22	39.88	56.89	69.58	67.73	34.74	64.77	—	15.68
FMSSD	89.11	**81.51**	48.22	**67.94**	69.23	73.56	76.87	90.71	**82.67**	73.33	52.65	67.52	72.37	**80.57**	**60.15**	**72.43**	16	11.74

TABLE VII
COMPARISON OF THE PERFORMANCE OF TWO-STAGE METHODS ON THE DOTA TESTING DATA SET

Method	Plane	BD	Bridge	GFT	SV	LV	Ship	TC	BC	ST	SBF	RA	Harbor	SP	HC	mAP↑	FPS↑	Std↓
R-FCN [5]	81.01	58.96	31.64	58.97	49.77	45.04	49.29	68.99	52.07	67.42	41.83	51.44	45.15	53.3	33.89	52.58	9	12.69
FR-H [4]	80.32	77.55	32.86	68.13	53.66	52.49	50.04	90.41	75.05	59.59	57.00	49.81	61.69	56.46	41.85	60.64	7	14.89
Deformable FR-H [47]	86.53	77.54	42.70	64.43	67.60	63.64	77.86	90.33	77.82	75.36	52.12	56.79	68.92	62.04	54.92	67.91	4	12.75
FPN [22]	88.70	75.10	52.60	59.20	69.40	**78.80**	**84.50**	90.60	81.30	82.60	52.50	62.10	76.70	66.30	60.10	72.00	6	12.27
ICN [25]	**90.00**	77.70	**53.40**	73.30	**73.50**	65.00	78.20	**90.80**	79.10	84.80	**57.20**	62.11	73.45	70.22	58.08	**72.45**	—	11.18
Yan et al. [26]	88.62	80.22	53.18	66.97	76.3	72.59	84.07	90.66	80.95	**76.24**	57.12	66.65	**74.08**	66.36	56.85	**72.72**	1	11.08
FMSSD	89.11	**81.51**	48.22	67.94	69.23	73.56	76.87	90.71	**82.67**	73.33	52.65	**67.52**	72.37	**80.57**	**60.15**	**72.43**	**16**	11.74

TABLE VIII
ARGUMENT OF THE WAY OF WEIGHT

Method	mAP
$2 - s_i$	70.50
$e^{-s_i} + 1$	**72.43**

TABLE IX
COMPARISON OF THE PERFORMANCE OF EACH DETECTION METHOD ON AIR-OBJ TESTING DATA SET

Method	TP	FP	Recall	Precision
Baseline	8057	459	90.40	94.61
Focal Loss	8196	351	91.96	95.89
+ASFP	8395	281	95.18	96.40
+Area-weighted Loss	8622	218	**96.74**	**97.53**

others. It is obvious that many small objects are in the class of bridge, SV, LV, and ship.

2) NWPU VHR-10 Data set: To verify the generalization of our method, we also product experiments on the NWPU VHR-10 data set, which is a ten-class geospatial object detection data set. These ten class of objects are airplane, ship, ST, BD, TC, BC, GFT, harbor, bridge, and vehicle. It contains 800 high-resolution remote sensing images cropped from Google Earth and Vaihingen data set and then manually annotated by experts [20]. We randomly select 60% of the original images as the training set, 20% as the validation set, and 20% as the testing set.

3) AIR-OBJ Data Set: The largest scale in the DOTA and NWPU VHR-10 data sets is $4k \times 4k$ pixels. Due to the lack of standard data sets of very large-scale remote sensing images, we collect 1853 airplane images and 1061 ship images that are 16 393 × 16 393 pixels, 0.6-m resolution. To increase the diversity of data, we collect images shot in multiple cities from Google Earth. We record the exact geographical coordinates of the location and capture the time of each image to ensure there are no duplicate images. All the images are processed in the same way as the DOTA data set and the NWPU VHR-10 data set. Finally, we obtain 17 042 pieces for training and 6260 pieces for testing. This data set will be published online soon.

B. Evaluation Metrics and Parameter Settings

To evaluate the ability of different methods in multiclass object detection for remote sensing imagery, mAP and FPS in PASCAL VOC are provided for data sets. Especially for multiscale objects, we adopt APs, APm, and APl to demonstrate the effectiveness of FMSSD. APs, APm, and APl mean the AP from 10 to 50 pixels, from 50 to 300 pixels, and above 300 pixels, respectively, which are similar to the metrics in COCO challenge [41]. To evaluate the significantly better results, we use the Tukey honest significant difference (HSD) posthoc test to calculate the standard deviation (Std) of each method [42].

The backbone network in our method is VGG-16 pretrained on the ILSVRC CLS-LOC data set [43]. All models are trained on the training set and tested on the testing set. In training and validating stages, we resize the sub-images to 512 × 512 pixels. Moreover, one image is randomly sampled in each mini-batch during training. When training, we use the "Xavier" method to randomly initialize the parameters in the extra added convolution layers and the stochastic gradient descent (SGD) method to optimize the model. Especially, the learning rate is 4e-4 for the first 150 epochs and then

TABLE X
COMPARISON OF THE PERFORMANCE OF OTHER METHODS ON NWPU VHR-10 DATA SET

Method	Plane	SH	ST	BD	TC	BC	GTF	Harbor	Bridge	Vehicle	mAP↑
Transferred CNN [20]	66.10	56.90	84.30	81.60	35.00	45.90	80.00	62.00	42.30	42.90	59.70
RICNN [20]	88.35	77.34	85.27	88.12	40.83	58.45	86.73	68.60	61.51	71.10	72.63
R-P-Faster R-CNN [48]	90.40	75.00	44.40	89.90	79.00	77.60	87.70	79.10	68.20	73.20	76.50
SSD512 [7]	90.40	60.90	79.80	89.90	82.60	80.60	98.30	73.40	76.70	52.10	78.40
DSSD321 [23]	86.50	65.40	90.30	89.60	85.10	80.40	78.20	70.50	68.20	74.20	78.80
DSOD300 [49]	82.70	62.80	89.20	90.10	87.80	80.90	79.80	82.10	81.20	61.30	79.80
R-FCN [5]	81.70	80.60	66.20	90.30	80.20	69.70	89.80	78.60	47.80	78.30	76.30
Deformable R-FCN [50]	87.30	81.40	63.60	90.40	81.60	74.10	90.30	75.30	71.40	75.50	79.10
Faster R-CNN [4]	94.60	82.30	65.32	95.50	81.90	89.70	92.40	72.40	57.50	77.80	80.90
Deformable Faster R-CNN [47]	90.70	87.10	70.50	89.50	89.30	87.30	97.20	73.50	69.90	**88.80**	84.40
RDAS512 [24]	99.60	85.50	89.00	95.00	89.60	94.80	95.30	82.60	77.20	86.50	89.50
Multi-Scale CNN [18]	99.30	**92.00**	83.20	97.20	**90.80**	92.60	98.10	**85.10**	71.90	85.90	89.60
FMSSD	**99.70**	89.90	**90.30**	**98.20**	86.00	**96.80**	**99.60**	75.60	**80.10**	88.20	**90.40**

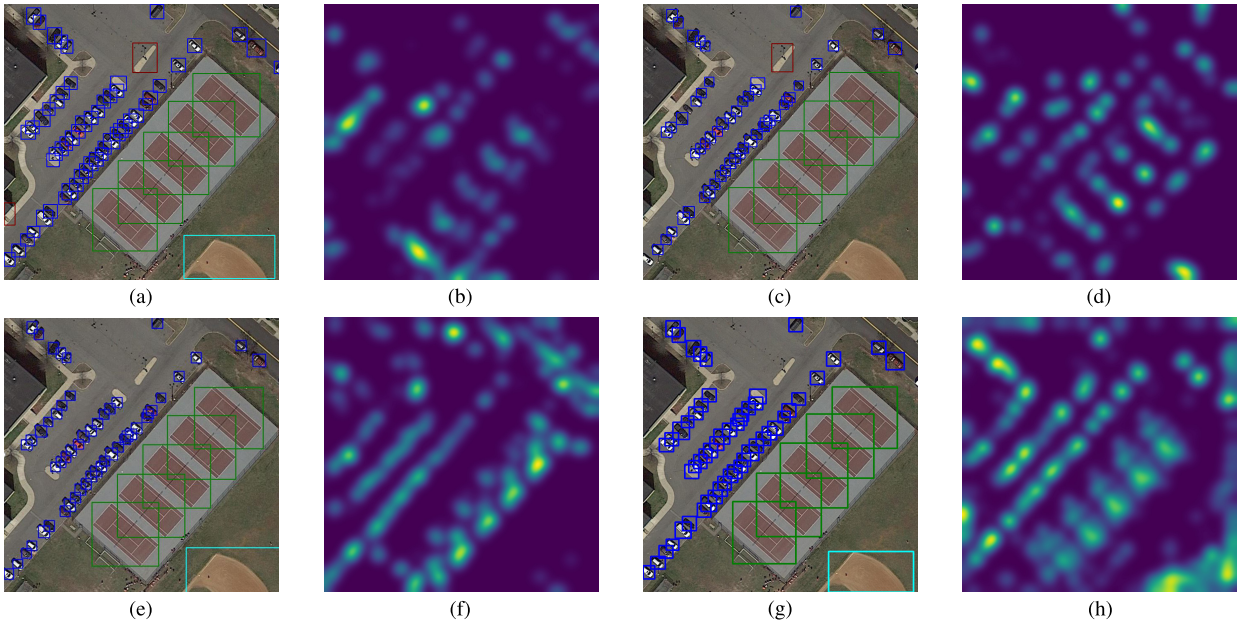


Fig. 8. Detection and visualize results for different methods on the DOTA data set. (a) and (b) Result of SSD-pretrained. (c) and (d) Result of adding focal loss. (e) and (f) Result of adding ASFP module. (g) and (h) Result of adding area-weighted loss.

decays to 4e-5 for latter epochs with batch size being 16, and the value of momentum and weight decay being 0.9 and 5e-4, respectively. We sample a batch of anchors, where the ratio of positive to negative samples is 1:3. At the validating and testing stage, we apply the nonmaximum suppression with the Jaccard overlap of 0.45 per class. Our baseline is the conventional SSD-pretrained based on PyTorch 0.4.0, whose source code is available. A total of 200 epochs are performed for our experiments, on two GeForce GTX 1080ti GPUs.

C. Quantitative Analysis

There are 937 high-resolution images in the testing set. In this section, we first show the different effectiveness of our method with comprehensive ablation experiments on the DOTA data set. Then, we show the comparison with the state-of-the-art methods on the DOTA data set and the NWPU VHR-10 data set. Tables III–X show the detection results on different data sets.

1) Ablation Experiments: We perform a series of ablation experiments on DOTA data set and the AIR-OBJ data set, so that our method achieves state-of-the-art performance. We use the same training data and parameter settings to ensure the fairness and accuracy of experiments. The results on the DOTA testing set are obtained by submitting the predictions to the official DOTA evaluation server, which only returns mAP. Hence, we calculate APs, APm, and APl on the validation set. All the speeds are tested on one image with a size of 1024 × 1024 pixels on a single GeForce GTX 1080Ti. Tables III, IV, IX, and Fig. 6 show the experimental results of various methods. We will analyze the difference and primary role of each structure in our experiments.

a) Baseline setup: In this article, we build the baseline network inspired by SSD-pretrained with the backbone VGG-16. It is obvious that the SSD-pretrained can detect objects in remote sensing imagery with fast speed 26 FPS. However, the

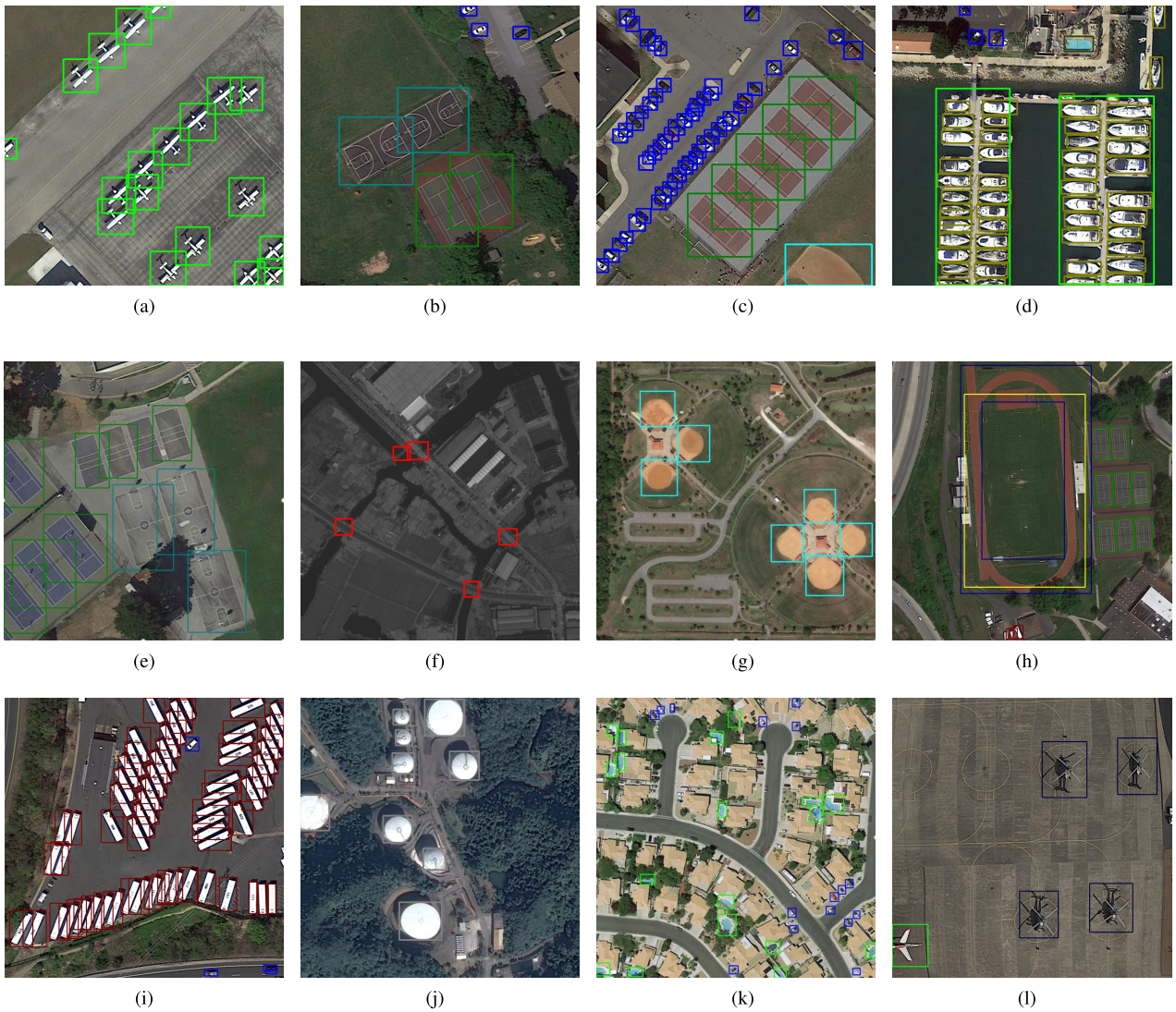


Fig. 9. Detection results for each class on the DOTA data set. Our method performs well on dense object detection, small object detection. (a) Plane. (b) TC and BC. (c) SV. (d) Ship and Harbor. (e) Baseball. (f) Bridge. (g) RA. (h) Ground track field. (i) LV. (j) ST. (k) SP. (l) HC.

detection performance of small objects is not satisfying. For instance, the mAP of the bridges is only 34.75%.

b) Effect of focal loss: There are many easy examples during training for one-stage detectors, which may lead the training to a bad result. We add the focal loss as the loss on the output of the classification subnet. Focal Loss can prevent easily classified examples from overwhelming the majority of the loss and dominating the gradient. As shown in Fig. 7, the focal loss decreases more quickly and the value is lower than the original classification loss. The final value of the focal loss is less than 1, which mainly consists of the classification loss of hard examples. Hence, the overall accuracy of detection has improved to 64.98% mAP.

c) Effect of ASFP module: It can be evidenced in Table III that the detection result has been improved by adding the ASFP module. Small objects have larger improvements with the ASFP module. For example, ships improve about 10% mAP than before. SVs, LVs, RA, SPs, and HCs also have great enhancement. We argue whether not all high-level features are helpful to small objects by some experiments. The results listed in Table V are obtained by starting a top-down structure from a different layer, which proves that the feature pyramid started from the last layer is more powerful.

d) Effect of area-weighted loss: We conduct three experiments based on area-weighted loss: only in classification, only in localization, and both of them. Experiments show that the last one can mostly enhance the accuracy of objects, especially for small objects, such as bridges, vehicles, and ships. As a result, we conclude that the area weight is helpful for not only classification but also localization. The loss can guide the network to learn the semantically strong information. As shown in Table IV, the performances of small objects have larger improvements with the ASFP module and Area-weighted loss function. The total mAP has increased by 3.50% with the area-weighted loss than before, and the inference speed has increased by 1 FPS. It is obvious in Fig. 6 that our method improves the performance of small object detection.

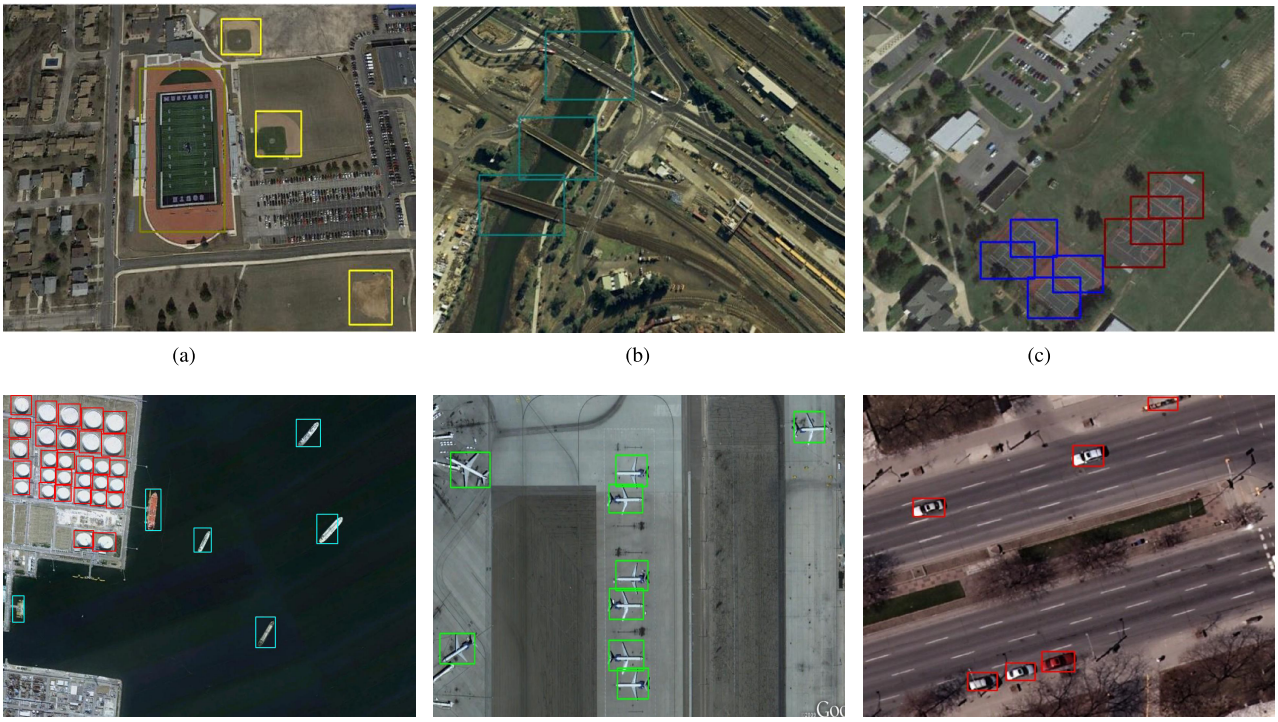


Fig. 10. Detection results for each class on the NWPU VHR-10 data set. (a) Ground track and Baseball. (b) Bridge. (c) TC and BC. (d) Ship and ST. (e) Airplane. (f) Vehicle.

Table VIII lists the arguments of the different ways of the weight. The exponential function has a better performance than the linear function.

We produce several experiments on the large-scale data set with the same experimental setting as the DOTA data set. Taking airplane detection for example, as shown in Table IX, it is obvious that our method can decrease false positives and improve detection accuracy effectively.

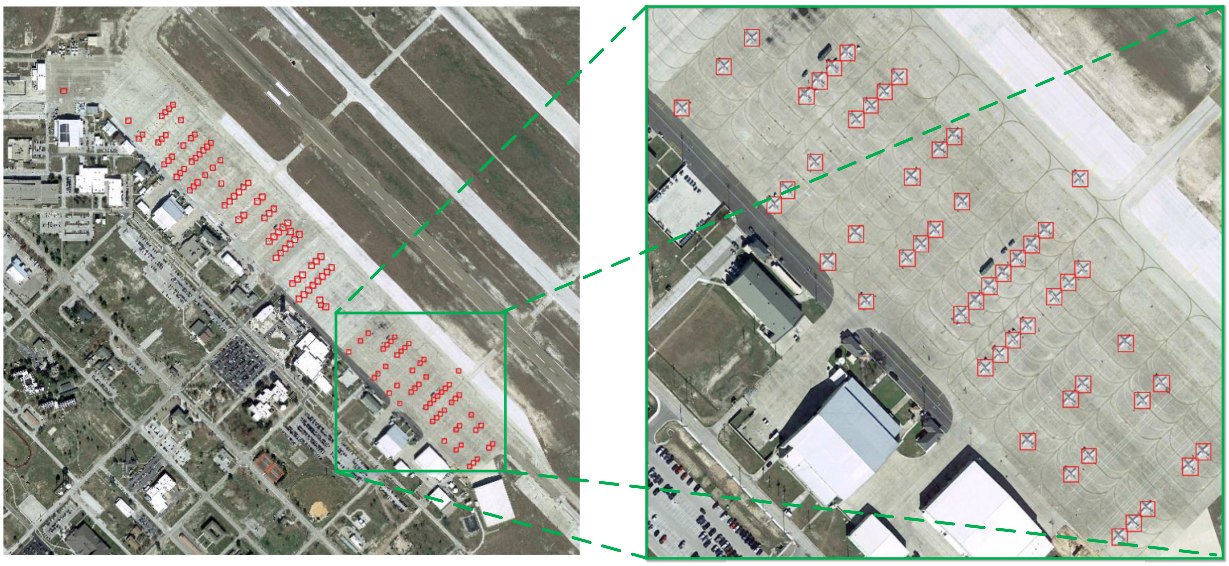
2) Comparison With the State-of-the-Art: Table VI and Table VII show the performances of our method and the state-of-the-art detectors on the HBB prediction task of the DOTA data set. Taking Std into consideration, the distribution of FMSSD is more uniform than most existing methods. And our method produces 72.43% mAP without bells and whistles, which is the first one-stage method achieving above 70% mAP in all published algorithms on the DOTA data set, surpassing all one-stage detectors, e.g., DSSD, YOLOv2, and RetinaNet. Compared to the two-stage methods, our method performs better than most of them except IoU-Adaptive Deformable R-CNN, which is based on ResNet-101 and uses images with a larger input size (800×800) [26]. And its inference speed is only 1 FPS. Huang *et al.* [51] prove that the input size plays an important role in detection accuracy because high-resolution inputs help the detectors "find" small objects. Due to the lack of computational resources, we only train our models with an input size of 512. The image pyramid can also be applied to our method to further improve the detection performance.

Tables III, VI, and VII present the inference speed of our method and other two-stage methods on the HBB prediction task of the DOTA data set. Although the ASFP module has a little influence on inference speed, our method is faster than all two-stage methods.

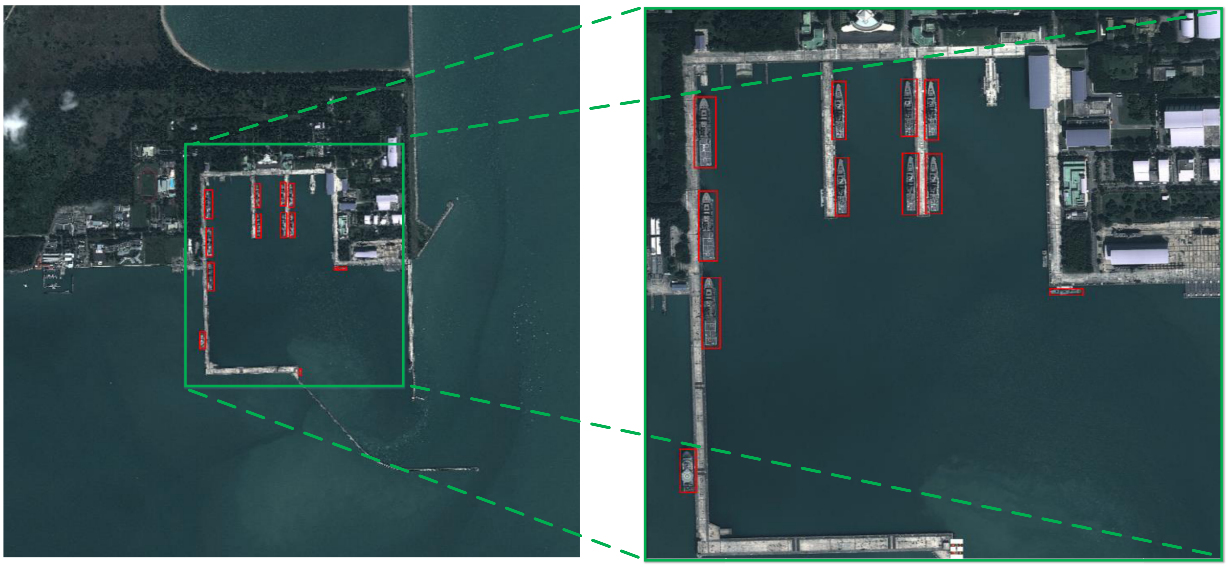
Table X shows the performance of our method and other detectors on the HBB prediction task of the NWPU VHR-10 data set. The same as the task of the DOTA data set, our method achieves state-of-the-art performance.

D. Qualitative Analysis

Fig. 8 shows the comparison of different methods on the DOTA data set. As shown in Fig. 8(a) and (b), the information in SSD-pretrained is not enough to accurately detect the objects. Fig. 8(c) and (d) also shows that the feature map obtains the information of hard examples. Fig. 8(e) and (f) shows more distinct context information of SVs. As presented in Fig. 8(h), the context information of objects is more abundant. The detection result of Fig. 8(h) can be seen in Fig. 8(g), which shows that the more the accurate prediction, the less the missed detection. With the improvement of our method, the context information of objects becomes more distinct. Moreover, we find that the boxes of objects are regressed more accurately. Fig. 9 shows the result of each class on the DOTA data set with our method. The detector has a good performance on multiclass objects with our method. Fig. 10 shows the detection result of each class on the NWPU VHR-10 data set. Fig. 11 shows the detection result of FMSSD in very large-scale remote sensing imagery. Hence, we can prove that our method has a great generalization performance on the task of multiclass object detection in large-scale remote sensing imagery.



(a) airplane



(b) ship

Fig. 11. Detection results for each class in very large-scale image. (a) Airplane. (b) Ship.

IV. Conclusion

In this article, we propose FMSSD for multiclass object detection in large-scale remote sensing imagery. In contrast to the widely employed detectors, FMSSD leverage context information both in multiscale feature maps and the same feature maps. Our method uses an ASFP module to obtain abundant context information on multiple scales. And we design a new area-weighted loss function which can guide the network to pay more attention to small objects in the same feature maps. Experiments based on the DOTA data set, NWPU VHR-10 data set, and AIR-OBJ data set demonstrate that our method outperforms the baseline method by a large margin and achieves the best trade-off between accuracy and speed in all published algorithms. For future work, we will pay more attention to the newly developed feature fusion networks and further improve our model.

Acknowledgment

The authors would like to thank all the reviewers for their valuable comments and feedback.

References

[1] A. Voulodimos *et al.*, "Deep learning for computer vision: A brief review," *Comput. Intell. Neurosci.*, vol. 2018, pp. 1–13, Jul. 2018.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[3] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[4] S. Ren, K. He, R. Girshick, and S. Jian, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[5] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[7] W. Liu *et al.*, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[8] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[9] J. Pang, C. Li, J. Shi, Z. Xu, and H. Feng, "R^2-CNN: Fast tiny object detection in large-scale remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5512–5524, Aug. 2019.

[10] Y. Long, Y. Gong, Z. Xiao, and Q. Liu, "Accurate object localization in remote sensing images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2486–2498, May 2017.

[11] Y. Zhang, Y. Yuan, Y. Feng, and X. Lu, "Hierarchical and robust convolutional neural network for very high-resolution remote sensing object detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5535–5548, Aug. 2019.

[12] X. Wu, D. Hong, J. Tian, J. Chanussot, W. Li, and R. Tao, "ORSIm detector: A novel object detection framework in optical remote sensing imagery using spatial-frequency channel features," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5146–5158, Jul. 2019.

[13] Z. Wang, L. Du, J. Mao, B. Liu, and D. Yang, "SAR target detection based on SSD with data augmentation and transfer learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 1, pp. 150–154, Jan. 2018.

[14] I. Ševo and A. Avramović, "Convolutional neural network based automatic object detection on aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 5, pp. 740–744, May 2016.

[15] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, "Vehicle detection in satellite images by hybrid deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 10, pp. 1797–1801, Oct. 2014.

[16] N. Ammour, H. Alhichri, Y. Bazi, B. Benjdira, N. Alajlan, and M. Zuair, "Deep learning approach for car detection in UAV imagery," *Remote Sens.*, vol. 9, no. 4, p. 312, 2017.

[17] W. Diao, X. Sun, X. Zheng, F. Dou, H. Wang, and K. Fu, "Efficient saliency-based object detection in remote sensing images using deep belief networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 2, pp. 137–141, Feb. 2016.

[18] W. Guo, W. Yang, H. Zhang, and G. Hua, "Geospatial object detection in high resolution satellite images based on multi-scale convolutional neural network," *Remote Sens.*, vol. 10, no. 1, p. 131, Jan. 2018.

[19] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.

[20] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[21] C. Wang, X. Bai, S. Wang, J. Zhou, and P. Ren, "Multiscale visual attention networks for object detection in VHR remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 310–314, Feb. 2019.

[22] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.

[23] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," Jan. 2017, *arXiv:1701.06659*. [Online]. Available: https://arxiv.org/abs/1701.06659

[24] S. Chen, R. Zhan, and J. Zhang, "Geospatial object detection in remote sensing imagery based on multiscale single-shot detector with activated semantics," *Remote Sens.*, vol. 10, no. 6, p. 820, 2018.

[25] S. M. Azimi, E. Vig, R. Bahmanyar, M. Körner, and P. Reinartz, "Towards multi-class object detection in unconstrained remote sensing imagery," in *Proc. Asian Conf. Comput. Vis.* Springer, 2018, pp. 150–165.

[26] J. Yan, H. Wang, M. Yan, W. Diao, X. Sun, and H. Li, "IoU-adaptive deformable R-CNN: Make full use of IoU for multi-class object detection in remote sensing imagery," *Remote Sens.*, vol. 11, no. 3, p. 286, 2019.

[27] P. Sun, G. Chen, G. Luke, and Y. Shang, "Salience biased loss for object detection in aerial images," Oct. 2018, *arXiv:1810.08103*. [Online]. Available: https://arxiv.org/abs/1810.08103

[28] Y. Yuan, Z. Xiong, and Q. Wang, "VSSA-NET: Vertical spatial sequence attention network for traffic sign detection," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3423–3434, Jul. 2019.

[29] P. Zhou, B. Ni, C. Geng, J. Hu, and Y. Xu, "Scale-transferrable object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 528–537.

[30] Q. Wang, M. Chen, F. Nie, and X. Li, "Detecting coherent groups in crowd scenes by multiview clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[31] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2403–2412.

[32] G.-S. Xia *et al.*, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983.

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, Apr. 2015, pp. 1–10.

[34] A. J. Collegio, J. C. Nah, P. S. Scotti, and S. Shomstein, "Attention scales according to inferred real-world object size," *Nature Hum. Behav.*, vol. 3, no. 1, pp. 40–47, 2019.

[35] Y. Hu, X. Li, N. Zhou, L. Yang, L. Peng, and S. Xiao, "A sample update-based convolutional neural network framework for object detection in large-area remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 6, pp. 947–951, Jun. 2019.

[36] S. Razakarivony and F. Jurie, "Vehicle detection in aerial imagery," *J. Vis. Commun. Image Represent.*, vol. 34, pp. 187–203, Jan. 2016.

[37] T. N. Mundhenk, G. Konjevod, W. A. Sakla, and K. Boakye, "A large contextual dataset for classification, detection and counting of cars with deep learning," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 785–800.

[38] K. Liu and G. Mattyus, "Fast multiclass vehicle detection on aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 9, pp. 1938–1942, Sep. 2015.

[39] H. Zhu, X. Chen, W. Dai, K. Fu, Q. Ye, and J. Jiao, "Orientation robust object detection in aerial images using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 3735–3739.

[40] Z. Liu, H. Wang, H. Weng, and L. Yang, "Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 8, pp. 1074–1078, Aug. 2016.

[41] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.

[42] E. Protopapadakis, D. Niklis, M. Doumpos, A. Doulamis, and C. Zopounidis, "Sample selection algorithms for credit risk modelling through data mining techniques," *Int. J. Data Mining, Model. Manage.*, vol. 11, no. 2, pp. 103–128, 2019.

[43] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[44] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7263–7271.

[45] O. Acatay, L. Sommer, A. Schumann, and J. Beyerer, "Comprehensive evaluation of deep learning based detection methods for vehicle detection in aerial imagery," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.

[46] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," Apr. 2018, *arXiv:1804.02767*. [Online]. Available: https://arxiv.org/abs/1804.02767

[47] Y. Ren, C. Zhu, and S. Xiao, "Deformable faster R-CNN with aggregating multi-layer features for partially occluded object detection in optical remote sensing images," *Remote Sens.*, vol. 10, no. 9, p. 1470, Sep. 2018.

[48] X. Han, Y. Zhong, and L. Zhang, "An efficient and robust integrated geospatial object detection framework for high spatial resolution remote sensing imagery," *Remote Sens.*, vol. 9, no. 7, p. 666, 2017.

[49] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "DSOD: Learning deeply supervised object detectors from scratch," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1937–1945.

[50] Z. Xu, X. Xin, L. Wang, R. Yang, and F. Pu, "Deformable convnet with aspect ratio constrained NMS for object detection in remote sensing imagery," *Remote Sens.*, vol. 9, no. 12, p. 1312, Dec. 2017.

[51] J. Huang *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3296–3297.

Peijin Wang (S'19) received the B.S. degree from Tianjin University, Tianjin, China, in 2017. She is currently pursuing the master's degree with the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, China.

Her research interests include computer vision and deep learning, especially on object detection, semantic segmentation, and remote sensing.

Xian Sun (SM'19) received the B.Sc. degree from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2004, and the M.Sc. and Ph.D. degrees from the Institute of Electronics, Chinese Academy of Sciences, China, in 2009.

He is currently a Professor with the Institute of Electronics, Chinese Academy of Sciences, China. His research interests include computer vision, geospatial data mining, and remote sensing image understanding.

Wenhui Diao (M'16) received the B.Sc. degree from Xidian University, Xi'an, China, in 2011, and the M.Sc. and Ph.D. degrees from the Institute of Electronics, Chinese Academy of Sciences, Beijing, China, in 2016.

He is currently an Assistant Professor with the Institute of Electronics, Chinese Academy of Sciences, China. His research interests include computer vision and remote sensing image analysis.

Kun Fu (M'16) received the B.Sc., M.Sc., and Ph.D. degrees from the National University of Defense Technology, Changsha, China, in 1995, 1999, and 2002, respectively.

He is currently a Professor with the Institute of Electronics, Chinese Academy of Sciences, Beijing, China. His research interests include computer vision, remote sensing image understanding, and geospatial data mining and visualization.